

BEST AVAILABLE COPY

PCT/JP2004/000384

日本国特許庁  
JAPAN PATENT OFFICE

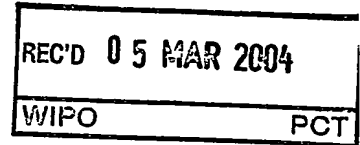
19.1.2004  
#

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日  
Date of Application: 2003年 1月21日

出願番号  
Application Number: 特願2003-011857  
[ST. 10/C]: [JP2003-011857]



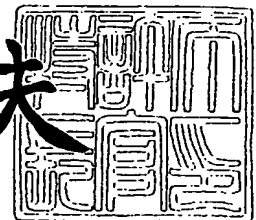
出願人  
Applicant(s): 三洋電機株式会社

PRIORITY DOCUMENT  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)

2004年 2月19日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



BEST AVAILABLE COPY

出証番号 出証特2004-3010908

【書類名】 特許願

【整理番号】 03A21P2877

【提出日】 平成15年 1月21日

【あて先】 特許庁長官殿

【国際特許分類】 G06T 11/00

【発明者】

【住所又は居所】 大阪府守口市京阪本通2丁目5番5号 三洋電機株式会社  
社内

【氏名】 郭 順也

【特許出願人】

【識別番号】 000001889

【氏名又は名称】 三洋電機株式会社

【代理人】

【識別番号】 100090181

【弁理士】

【氏名又は名称】 山田 義人

【手数料の表示】

【予納台帳番号】 014812

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

## 【書類名】

明 書

## 【発明の名称】

メニュー表示制御データのデータ構造

## 【特許請求の範囲】

## 【請求項 1】

メニュー画像を表示する表示装置のプロセサによって読み取られるメニュー表示制御データのデータ構造において、

前記プロセサによって表示処理を施される複数の大項目をそれぞれ管理する複数の第 1 管理テーブル、および

前記複数の第 1 管理テーブルの各々に属するかつ各自が前記プロセサによって表示処理を施される複数の小項目を管理する複数の第 2 管理テーブルを備え、

注目する第 2 管理テーブルが属する第 1 管理テーブルと異なる第 1 管理テーブルの下で管理される小項目に依存していることを示す依存関係情報を前記注目する第 2 管理テーブルに割り当て、所望の小項目が選択されたときに前記プロセサが前記依存関係情報に基づいて前記所望の小項目に依存する複数の小項目を表示できるようにしたことを特徴とする、データ構造。

## 【請求項 2】

前記所望の小項目を管理する第 1 管理テーブルに前記所望の小項目を示す所望小項目情報を割り当て、前記プロセサが前記所望小項目情報に基づいて前記所望の小項目に対応する大項目の代わりに前記所望の小項目を表示できるようにした、請求項 1 記載のデータ構造。

## 【請求項 3】

前記所望の小項目の選択が解除されたとき前記所望小項目情報を小項目未選択情報に切り換え、前記プロセサが前記小項目未選択情報に基づいて前記所望の小項目の代わりに前記所望の小項目に対応する大項目を表示できるようにした、請求項 2 記載のデータ構造。

## 【請求項 4】

前記所望の小項目が選択されていない第 2 管理テーブルの小項目に依存する第 2 管理テーブルが属する第 1 管理テーブルに選択不可情報を割り当て、前記プロセサが、前記選択不可情報が割り当てられた第 1 管理テーブルが管理する大項目

の表示を中止できるようにした、請求項1ないし3のいずれかに記載のデータ構造。

#### 【請求項5】

前記複数の第1管理テーブルの各々に属する複数の第2管理テーブルは連続し、前記複数の第1管理テーブルの各々には前記複数の第2管理テーブルの先頭位置情報とテーブル数情報とが割り当てられる、請求項1ないし4のいずれかに記載のデータ構造。

#### 【請求項6】

前記依存関係情報は複数の小項目との依存関係を示しうる、請求項1ないし5のいずれかに記載のデータ構造。

#### 【請求項7】

請求項1ないし6のいずれかに記載のメニュー表示制御データを格納するデータファイル。

#### 【発明の詳細な説明】

##### 【0001】

##### 【発明の属する技術分野】

この発明は、メニュー表示制御データのデータ構造に関し、特にたとえばメニュー画像を表示する表示装置のプロセサによって読み取られる、メニュー表示制御データのデータ構造に関する。

##### 【0002】

##### 【従来技術】

この種の表示装置では、メニュー表示制御データおよびこれに適合する制御プログラムは、製造段階において内蔵メモリに書き込まれる。このため、使用段階でメニュー表示制御データのみを変更すると、制御プログラムは変更されたメニュー表示制御データを適切に処理できず、メニュー表示に不具合が生じる。かかる理由から、従来は、使用段階でメニュー画像を変更することはできなかった。

##### 【0003】

##### 【発明が解決しようとする課題】

しかし、表示装置の使用状況の変化に対応してメニュー画像を変更できなけれ

ば、表示装置の汎用性に欠け、コストの面で問題が生じる。たとえば、撮影現場の詳細情報をメニュー操作によって入力できるデジタルカメラを想定した場合、被写界が楽器の製作現場であるときと家屋の建築現場であるときとで、入力すべき詳細情報が異なる。かかる場合にメニュー画像が固定であると、撮影現場毎に専用のデジタルカメラを準備しなければならず、コストがかかってしまう。

#### 【0004】

それゆえに、この発明の主たる目的は、異なるメニュー画像を表示装置に表示させることができる、メニュー表示制御データのデータ構造を提供することである。

#### 【0005】

##### 【課題を解決するための手段】

この発明によれば、メニュー画像を表示する表示装置のプロセサによって読み取られるメニュー表示制御データのデータ構造は、プロセサによって表示処理を施される複数の大項目をそれぞれ管理する複数の第1管理テーブル、および複数の第1管理テーブルの各々に属するかつ各自がプロセサによって表示処理を施される複数の小項目を管理する複数の第2管理テーブルを備え、注目する第2管理テーブルが属する第1管理テーブルと異なる第1管理テーブルの下で管理される小項目との依存関係を示す依存関係情報を注目する第2管理テーブルに割り当て、所望の小項目が選択されたときにプロセサが依存関係情報に基づいて所望の小項目に依存する複数の小項目を表示できるようにしたことを特徴とする。

#### 【0006】

##### 【作用】

表示装置のプロセサによって読み取られるメニュー表示制御データは、複数の第1管理テーブルとその各々に属する複数の第2管理テーブルとを備える。複数の第1管理テーブルは、プロセサによって表示処理を施される複数の大項目をそれぞれ管理する。また、各々の第2管理テーブルは、プロセサによって表示処理を施される複数の小項目を管理する。

#### 【0007】

注目する第2管理テーブルには、当該第2管理テーブルが属する第1管理テ

ブルと異なる第1管理テーブルの下で管理される小項目との依存関係を示す依存関係情報が割り当てられる。所望の小項目が選択されると、プロセサは、この依存関係情報に基づいて、所望の小項目に依存する複数の小項目を表示する。

#### 【0008】

このように、依存関係情報が第2管理テーブルに割り当てられるため、プロセサは、この依存関係情報を読み取ることで小項目を的確に画面に表示することができる。つまり、表示装置は異なるメニュー画像を共通の手順で画面に表示することができる。

#### 【0009】

好ましくは、所望の小項目を示す所望小項目情報が、所望の小項目を管理する第1管理テーブルに割り当てられる。プロセサは、かかる所望小項目情報に基づいて、所望の小項目に対応する大項目の代わりに所望の小項目を表示する。これによって、いずれの小項目が選択されたかを容易に把握することができ、操作性が向上する。

#### 【0010】

さらに好ましくは、所望の小項目の選択が解除されたとき、所望小項目情報が小項目未選択情報に切り換えられる。プロセサは、小項目未選択情報に基づいて、所望の小項目の代わりに所望の小項目に対応する大項目を表示する。

#### 【0011】

好ましくは、所望の小項目が選択されていない第2管理テーブルの小項目に依存する第2管理テーブルが属する第1管理テーブルに選択不可情報が割り当てられる。プロセサは、選択不可情報が割り当てられた第1管理テーブルが管理する大項目の表示を中止する。これによって、未選択の小項目に関連する大項目の表示が中止され、操作性が向上する。

#### 【0012】

好ましくは、依存関係情報は複数の小項目との依存関係を示しうる。この場合、第2管理テーブルの数が減少され、データサイズの抑制が可能となる。

#### 【0013】

好ましくは、複数の第1管理テーブルの各々に属する複数の第2管理テーブル

は連続する。このとき、複数の第1管理テーブルの各々には、複数の第2管理テーブルの先頭位置情報とテーブル数情報とが割り当てられる。これによって、第2管理テーブル毎にアドレス情報を割り当てる必要がなくなり、データサイズの抑制が可能となる。

#### 【0014】

##### 【発明の効果】

この発明によれば、依存関係情報が第2管理テーブルに割り当てられるため、プロセサは、この依存関係情報を読み取ることで小項目を的確に画面に表示することができる。つまり、表示装置は異なるメニュー画像を共通の手順で画面に表示することができる。

#### 【0015】

この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

#### 【0016】

##### 【実施例】

図1を参照して、この実施例のデジタルカメラ10は、主に楽器の製作現場を撮影するために用いられるカメラであり、イメージセンサ12を含む。被写界つまり製作現場の光学像は、光学レンズ（図示せず）を介してイメージセンサ12の受光面に入射する。受光面には複数の受光素子（図示せず）が配置され、かかる受光素子の光電変換によって、被写界の光学像に対応する生画像信号（電荷）が生成される。

#### 【0017】

シャッターキー46が操作されると、撮影処理が実行される。まず、CPU44が、TG（Timing Generator）14に読み出し命令を与え、信号処理回路18に処理命令を与え、そしてJPEGコーデック24に圧縮命令を与える。TG14は、受光素子で生成された生画像信号を読み出すべく、タイミング信号をイメージセンサ12に与える。生画像信号は、ラスタスキャン方式でイメージセンサ12から読み出され、CDS/AGC/AD回路16でノイズ除去、ゲイン調整およびA/D変換の一連の処理を施される。CDS/AGC/AD回路16から出

力された生画像データは、信号処理回路18によって色分離、白バランス調整、YUV変換などの信号処理を施される。これによって生成されたYUVデータは、メモリコントローラ20によってSDRAM22に書き込まれる。

#### 【0018】

JPEGコーデック24は、メモリコントローラ20を通してSDRAM22からYUVデータを読み出し、読み出されたYUVデータにJPEG圧縮を施す。これによって生成された圧縮YUVデータつまりJPEGデータもまた、メモリコントローラ20によってSDRAM22に書き込まれる。

#### 【0019】

CPU44は、自らファイルヘッダを作成し、SDRAM22に格納されたJPEGデータをメモリコントローラ20を通して読み出し、ファイルヘッダとJPEGデータとを含む画像ファイルをI/F34に与える。メモリカード38がスロット36に装着されている場合、画像ファイルは、I/F34によってメモリカード38に記録される。通信カード40がスロット36に装着されている場合、画像ファイルは、I/F34から通信カード40に与えられ、通信カード40によって遠隔地のサーバ（図示せず）に送信される。

#### 【0020】

後述する表示制御テーブルおよび文字列データが有効である場合、メニューキー46およびセットキー50の操作に応答して、図2～図4に示す構造を有するメニューがLCD30に表示される。具体的には、CPU44からの命令に従って、キャラクタジェネレータ32から所望のキャラクタデータが出力される。出力されたキャラクタデータはミキサ28を介してLCD30に与えられ、これによってメニューが表示される。なお、メニュー構造を形成する各要素の名称を図5に示す。また、キー操作に応じた表示メニューの遷移を図6～図14（B）に示す。かかる表示メニューを操作することで、楽器の製作現場を特定するための詳細情報の入力が可能となる。

#### 【0021】

図2～図4を参照して、メニュー構造は、“音楽ジャンル”，“楽器分類”，“楽器名”，“材質”，“材料”および“表面処理”の6つの大項目に分類され



る。この6つの大項目には、“0”～“5”の大項目番号が割り当てられる。

#### 【0022】

“音楽ジャンル”は、“クラシック”、“ジャズ”および“その他”の小項目に分類される。いずれの小項目も、枝メニュー“0”に属する。“クラシック”には小項目番号“0”が割り当てられ、“ジャズ”には小項目番号“1”が割り当てられ、そして“その他”には小項目番号“2”が割り当てられる。

#### 【0023】

“楽器分類”は、“弦楽器”、“木管楽器”、“金管楽器”、“打楽器”および“鍵盤楽器”の小項目に分類される。これらの小項目も、全て枝メニュー“0”に属する。“弦楽器”には小項目番号“0”が割り当てられ、“木管楽器”には小項目番号“1”が割り当てられ、“金管楽器”には小項目番号“2”が割り当てられ、“打楽器”には小項目番号“3”が割り当てられ、そして“鍵盤楽器”には小項目番号“4”が割り当てられる。

#### 【0024】

“楽器名”は、“バイオリン”、“ビオラ”、“チェロ”、“コントラバス”、“オーボエ”、“フルート”、“クラリネット”、“ファゴット”、“サクソフォン”、“ホルン”、“トランペット”、“トロンボーン”、“ユーフォニアム”、“チューバ”、“ティンパニ”、“スネアドラム”、“バスドラム”、“ピアノ”、“チェレスタ”および“オルガン”の小項目に分類される。

#### 【0025】

このうち、“バイオリン”、“ビオラ”、“チェロ”および“コントラバス”は枝メニュー“0”に属し、“オーボエ”、“フルート”、“クラリネット”、“ファゴット”および“サクソフォン”は枝メニュー“1”に属する。また、“ホルン”、“トランペット”、“トロンボーン”、“ユーフォニアム”および“チューバ”は枝メニュー“2”に属し、“ティンパニ”、“スネアドラム”および“バスドラム”は枝メニュー“3”に属し、“ピアノ”、“チェレスタ”および“オルガン”は枝メニュー“4”に属する。

#### 【0026】

“バイオリン”には小項目番号“0”が割り当てられ、“ビオラ”には小項目

番号“1”が割り当てられ、“チェロ”には小項目番号“2”が割り当てられ、そして“コントラバス”には小項目番号“3”が割り当てられる。“オーボエ”には小項目番号“0”が割り当てられ、“フルート”には小項目番号“1”が割り当てられ、“クラリネット”には小項目番号“2”が割り当てられ、“ファゴット”には小項目番号“3”が割り当てられ、そして“サクソフォン”には小項目番号“4”が割り当てられる。

**【0027】**

“ホルン”には小項目番号“0”が割り当てられ、“トランペット”には小項目番号“1”が割り当てられ、“トロンボーン”には小項目番号“2”が割り当てられ、“ユーフォニアム”には小項目番号“3”が割り当てられ、“チューバ”には小項目番号“4”が割り当てられる。“ティンパニ”には小項目番号“0”が割り当てられ、“スネアドラム”には小項目番号“1”が割り当てられ、“バスクドラム”には小項目番号“2”が割り当てられる。“ピアノ”には小項目番号“0”が割り当てられ、“チェレスタ”には小項目番号“1”が割り当てられ、“オルガン”には小項目番号“2”が割り当てられる。

**【0028】**

“材質”は、“木材”、“金属”および“その他”の小項目に分類される。“木材”は、枝メニュー“0”および“1”に属し、“金属”は枝メニュー“1”および“2”に属し、“その他”は枝メニュー“0”、“1”および“2”に属する。枝メニュー“0”に属する“木材”には小項目番号“0”が割り当てられ、枝メニュー“1”に属する“木材”には小項目番号“1”が割り当てられる。枝メニュー“1”に属する“金属”には小項目番号“0”が割り当てられ、枝メニュー“2”に属する“金属”には小項目番号“0”が割り当てられる。枝メニュー“0”に属する“その他”には小項目番号“1”が割り当てられ、枝メニュー“1”に属する“その他”には小項目番号“2”が割り当てられ、枝メニュー“2”に属する“その他”には小項目番号“1”が割り当てられる。

**【0029】**

“材料”は、“真鍮（銅70％）”、“真鍮（銅80％）”、“真鍮（銅90％）”、“金”、“銀”、“マホガニ（紅木）”、“エボニ（黒檀）”、“ロー

ズウッド（紫檀）” および “その他” の小項目に分類される。“真鍮（銅 70 %）”，“真鍮（銅 80 %）”，“真鍮（銅 90 %）”，“金” および “銀” は、枝メニュー “0” に属する。“マホガニ（紅木）”，“エボニ（黒檀）” および “ローズウッド（紫檀）” は、枝メニュー “1” に属する。“その他” は、枝メニュー “0” および “1” に属する。

#### 【0030】

“真鍮（銅 70 %）” には小項目番号 “0” が割り当てられ、“真鍮（銅 80 %）” には小項目番号 “1” が割り当てられ、“真鍮（銅 90 %）” には小項目番号 “2” が割り当てられる。“金” には小項目番号 “3” が割り当てられ、“銀” には小項目番号 “4” が割り当てられる。枝メニュー “0” に属する “その他” には小項目番号 “5” が割り当てられる。“マホガニ（紅木）” には小項目番号 “0” が割り当てられ、“エボニ（黒檀）” には小項目番号 “1” が割り当てられ、“ローズウッド（紫檀）” には小項目番号 “2” が割り当てられる。枝メニュー “1” に属する “その他” には小項目番号 “3” が割り当てられる。

#### 【0031】

“表面処理” は、“金メッキ”，“銀メッキ”，“その他メッキ”，“漆”，“ニス”，“無垢”，“ラッカー” および “その他塗装” の小項目に分類される。“金メッキ”，“銀メッキ”，“その他メッキ” は枝メニュー “0” に属し、“漆” および “ニス” は枝メニュー “1” に属する。“無垢”，“ラッカー” および “その他塗装” の各々は、枝メニュー “0” および “1” に属する。

#### 【0032】

“金メッキ” には小項目番号 “3” が割り当てられ、“銀メッキ” には小項目番号 “4” が割り当てられ、“その他メッキ” には小項目番号 “5” が割り当てられる。“漆” には小項目番号 “2” が割り当てられ、“ニス” には小項目番号 “3” が割り当てられる。枝メニュー “0” および “1” のいずれにおいても、“無垢” には小項目番号 “0” が割り当てられ、“ラッカー” には小項目番号 “1” が割り当てられる。これに対して、枝メニュー “0” に属する “その他塗装” には小項目番号 “2” が割り当てられ、枝メニュー “1” に属する “その他塗装” には小項目番号 “4” が割り当てられる。

## 【0033】

各々の枝メニューにはまた、他の小項目との依存関係を示す3つの番号が割り当てられる。このうち、左端の番号は依存する大項目の番号であり、中央の番号は依存する枝メニューの番号であり、右端の番号は依存する小項目の番号である。なお、“-1”は依存関係がないことを意味する。したがって、“音楽ジャンル”または“楽器分類”に属する枝メニュー“0”は、いずれの小項目にも依存しない。

## 【0034】

これに対して、“楽器名”については、枝メニュー“0”が“楽器分類”の“弦楽器”に依存し、枝メニュー“1”が“楽器分類”の“木管楽器”に依存し、枝メニュー“2”が“楽器分類”の“金管楽器”に依存し、枝メニュー“3”が“楽器分類”の“打楽器”に依存し、そして枝メニュー“4”が“楽器分類”の“鍵盤楽器”に依存する。

## 【0035】

“材質”については、枝メニュー“0”が、“楽器分類”の“バイオリン”，“ビオラ”，“チェロ”，“コントラバス”，“オーボエ”，“クラリネット”または“ファゴット”に依存する。また、枝メニュー“1”が、“楽器分類”の“フルート”に依存する。さらに、枝メニュー“2”が、“サクソフォン”，“ホルン”，“トランペット”，“トロンボーン”，“ユーフォニアム”または“チューバ”に依存する。

## 【0036】

“材料”については、枝メニュー“0”が“材質”の“金属”に依存し、枝メニュー“1”が“材質”の“木材”に依存する。“表面処理”については、枝メニュー“0”が“材質”の“金属”に依存し、枝メニュー“1”が“材質”の“木材”に依存する。

## 【0037】

メニュー表示がオフされている状態でメニューキー48が操作されると、まず図7(A)に示す大項目メニューがLCD30に表示される。図7(A)によれば、“音楽ジャンル”および“楽器分類”の文字列が画面中央に表示され、“音

“音楽ジャンル”がカーソルCSによって指向される。カーソルCSは、アップキー52の操作に応答して上方に移動し、ダウンキー54の操作に応答して下方に移動する。なお、この時点で“音楽ジャンル”および“楽器分類”以外の大項目が表示されないのは、上述のような依存関係から、“楽器分類”に属する小項目を選択しない限り“楽器名”に関する小項目の選択を許容するべきではなく、“楽器名”に属する小項目を選択しない限り“材質”、“材料”または“表面処理”に関する小項目の選択を許容するべきではないからである。

#### 【0038】

図7(A)に示す状態でダウンキー54が1回操作されると、カーソルCSは、図7(B)に示すように“楽器分類”を指向する。カーソルCSが“楽器分類”を指向する状態でセットキー50が操作されると、画面表示は図7(B)に示す大項目メニューから図8(A)に示す枝メニューに遷移する。図8(A)によれば、“楽器分類”に属する小項目を表す“弦楽器”、“木管楽器”、“金管楽器”、“打楽器”および“鍵盤楽器”の文字列が、画面中央に表示される。カーソルCSは、“弦楽器”を指向する。この状態でダウンキー54が1回操作されると、カーソルCSの指向先は、図8(B)に示すように“木管楽器”に移行する。

#### 【0039】

カーソルCSが“木管楽器”を指向している状態でセットキー50が操作されると、画面表示は図8(B)に示す枝メニューから図9(A)に示す大項目メニューに遷移する。図9(A)によれば、“楽器分類”の文字列に代えて“木管楽器”の文字列が表示され、その下に“楽器名”の文字列が表示される。これによって、“楽器分類”として“木管楽器”が選択されたこと、ならびに“楽器名”に属する小項目の選択が促されていることが分かる。この時点では“楽器名”に属する小項目が特定されていないため、“材質”、“材料”および“表面処理”の大項目は未だ表示されない。なお、図8(B)に示す枝メニューが表示されているときにメニューキー48を操作すると、画面表示が図7(B)に示す大項目メニューに戻る。

#### 【0040】

ダウンキー 54 によってカーソル CS を“音楽ジャンル”から“楽器名”に合わせた状態でセットキー 50 を操作すると、画面表示は、図 9 (B) に示す大項目メニューから図 10 (A) に示す枝メニューに遷移する。図 10 (A) によれば、“楽器名”の枝メニュー“0”に属する“オーボエ”，“フルート”，“クラリネット”，“ファゴット”および“サクソフォン”の文字列が画面中央に表示される。かかる文字列が表示されるのは、図 2 に示す“木管楽器”との依存関係を示す“1”，“0”および“1”の番号が枝メニュー“0”に割り当てられているからである。なお、カーソル CS は、“オーボエ”を指向する。

#### 【0041】

この状態からダウンキー 54 によってカーソル CS を“フルート”に合わせ、かつセットキー 50 を操作すると、画面表示は図 10 (B) に示す枝メニューから図 11 (A) に示す大項目メニューに遷移する。図 11 (A) によれば、“楽器名”の文字列に代えて“フルート”の文字列が表示され、その下に“材質”，“材料”および“表面処理”の文字列が表示される。これによって、“楽器名”として“フルート”が選択されたこと、および“材質”，“材料”および“表面処理”の各々に属する小項目の選択が促されていることが分かる。なお、セットキー 50 の代わりにメニューキー 48 が押された場合、画面表示は図 10 (B) に示す枝メニューから図 9 (B) に示す大項目メニューに戻る。

#### 【0042】

ダウンキー 54 によってカーソル CS を“材質”に合わせた状態でセットキー 50 を操作すると、画面表示は、図 11 (B) の大項目メニューから図 12 (A) に示す枝メニューに遷移する。図 12 (A) によれば、“材質”に属する小項目を表す“金属”，“木材”および“その他”の文字列が画面中央に表示される。これらの小項目が表示されるのは、図 2 に示す“フルート”との依存関係を示す“2”，“1”および“1”の番号が図 3 に示す“金属”，“木材”および“その他”に割り当てられているからである。

#### 【0043】

ここで、ダウンキー 54 によってカーソル CS を“木材”に合わせ、かつセットキー 50 を操作すると、画面表示は、図 12 (B) に示す枝メニューから図 1

3 (A) に示す大項目メニューに遷移する。図 13 (A) によれば、“材質”の文字列に代えて“木材”の文字列が表示される。これによって、“材質”として“木材”が選択されたことが分かる。なお、セットキー 50 の代わりにメニューキー 48 が押された場合、画面表示は図 12 (B) に示す枝メニューから図 11 (B) に示す大項目メニューに戻る。

#### 【0044】

ダウンキー 54 によってカーソル CS を“材料”に合わせ、かつセットキー 50 を操作すると、画面表示は、図 13 (B) に示す大項目メニューから図 14 (A) に示す枝メニューに遷移する。図 14 (A) によれば、“材料”に属する小項目を表す“マホガニ (紅木)”, “エボニ (黒檀)”, “ローズウッド (紫檀)” および“その他”の文字列が画面中央に表示される。これらの小項目が表示されるのは、図 3 に示す“木材”との依存関係を示す“3”, “1” および “1” の番号が図 4 に示す“マホガニ (紅木)”, “エボニ (黒檀)”, “ローズウッド (紫檀)” および“その他”に割り当てられているからである。

#### 【0045】

図 14 (A) に示す枝メニューでは、カーソル CS は“マホガニ (紅木)”を指向する。ここで、ダウンキー 54 が操作されると、カーソル CS は、図 14 (B) に示すように“エボニ (黒檀)”に移動する。かかるキー操作を繰り返すことによって、各々の大項目について所望の小項目が選択されていく。なお、全ての大項目について小項目を選択する必要はない。選択が行なわれなかった大項目は、空欄とされる。

#### 【0046】

このような詳細情報の入力操作は、シャッターキー 46 の操作に先立って行われる。入力された詳細情報は、シャッターキー 46 の操作に応答して作成された画像ファイルのヘッダに埋め込まれる。これによって、どのような楽器が製作されているかを撮影画像と詳細情報とに基づいて正確に把握することができる。

#### 【0047】

続いて、詳細情報の入力に利用される表示制御テーブルおよび文字列データの構造を、図 15～図 17 を参照して説明する。表示制御テーブル GUICONF0.TBL は

、図15および図16に示す構造を有し、文字列データGUICONF0.DATは、図17に示す構造を有する。

#### 【0048】

図15および図16を参照して、表示制御テーブルGUICONF0.TBLは、1つのLANテーブル<lantable>[0]、および連続するN個のGUIテーブル<guitbl>[0]～<guitbl>[N-1]を含む。GUIテーブル<guitbl>[0]～<guitbl>[N-1]の各々には、連続するI個の枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[I-1]が割り当てられる。さらに、枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[I-1]の各々には、連続するK個のメニューストリング<menu\_\_str>[0]～<menu\_\_str>[K-1]と、連続するL個のツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[L-1]とが割り当てられる。

#### 【0049】

GUIテーブルは、図2～図4に示す大項目毎に用意される。上述のように、“音楽ジャンル”，“楽器分類”，“楽器名”，“材質”，“材料”および“表面処理”の6つが大項目として存在する。このため、GUIテーブル<guitbl>[0]～<guitbl>[5]が表示制御テーブルGUICONF0.TBLに用意される。

#### 【0050】

枝メニューテーブルは、図2～図4に示す枝メニュー毎に用意される。“音楽ジャンル”の枝メニューは1つであるため、枝メニューテーブル<menu\_\_tbl>[0]のみがGUIテーブル<guitbl>[0]に割り当てられる。“楽器分類”の枝メニューもまた1つであるため、GUIテーブル<guitbl>[1]にも枝メニューテーブル<menu\_\_tbl>[0]のみが割り当てられる。“楽器名”の枝メニューは4つであるため、GUIテーブル<guitbl>[2]には枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[3]が割り当てられる。“材質”の枝メニューは3つであるため、GUIテーブル<guitbl>[3]には枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[2]が割り当てられる。“材料”の枝メニューは2つであるため、GUIテーブル<guitbl>[4]には枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[1]が割り当てられる。“表面処理”の枝メニューもまた2つであるため、GUIテーブル<GUI TBL>[5]には枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[1]が割り当てら



れる。

#### 【0051】

メニューストリングは、図2～図4に示す小項目毎に用意される。“音楽ジャンル”については、3つの小項目“クラシック”，“ジャズ”および“その他”が枝メニュー“0”に属する。このため、メニューストリング<menu\_str>[0]～<menu\_str>[2]が、GUIテーブル<guitbl>[0]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられる。“楽器分類”については、4つの小項目が“弦楽器”，“木管楽器”，“金管楽器”，“打楽器”および“鍵盤楽器”が枝メニュー“0”に属する。このため、メニューストリング<menu\_str>[0]～<menu\_str>[3]が、GUIテーブル<guitbl>[1]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられる。

#### 【0052】

“楽器名”については、“バイオリン”，“ビオラ”，“チェロ”および“コントラバス”の4つの小項目が枝メニュー“0”に属し、“オーボエ”，“フルート”，“クラリネット”，“ファゴット”および“サクソフォン”の5つの小項目が枝メニュー“1”に属する。また、“ホルン”，“トランペット”，“トロンボーン”，“ユーフォニアム”，“チューバ”の5つの小項目が枝メニュー“2”に属し、“ティンパニ”，“スネアドラム”および“バストラム”の3つの小項目が枝メニュー“3”に属し、そして“ピアノ”，“チェレスタ”および“オルガン”の3つの小項目が枝メニュー“4”に属する。

#### 【0053】

このため、メニューストリング<menu\_str>[0]～<menu\_str>[3]がGUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられ、メニューストリング<menu\_str>[0]～<menu\_str>[4]がGUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[1]に割り当てられる。また、メニューストリング<menu\_str>[0]～<menu\_str>[4]がGUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[2]に割り当てられ、メニューストリング<menu\_str>[0]～<menu\_str>[2]がGUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[3]に割り当てられ、メニューストリング<menu\_str>

>[0]～<menu\_str>[2]がGUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[4]に割り当てられる。

#### 【0054】

“材質”については、“木材”および“その他”の2つの小項目が枝メニュー“0”に属し、“金属”，“木材”および“その他”の3つの小項目が枝メニュー“1”に属し、そして“金属”および“その他”の2つの小項目が枝メニュー“2”に属する。このため、メニューストリング<menu\_str>[0]～<menu\_str>[1]がGUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられ、メニューストリング<menu\_str>[0]～<menu\_str>[2]がGUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_tbl>[1]に割り当てられ、そしてメニューストリング<menu\_str>[0]～<menu\_str>[1]がGUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_tbl>[2]に割り当てられる。

#### 【0055】

“材料”については、“真鍮（銅70％）”，“真鍮（銅80％）”，“真鍮（銅90％）”，“金”，“銀”および“その他”の6つの小項目が枝メニュー“0”に属し、“マホガニ（紅木）”，“エボニ（黒檀）”，“ローズウッド（紫檀）”および“その他”の4つの小項目が枝メニュー“1”に属する。このため、メニューストリング<menu\_str>[0]～<menu\_str>[5]がGUIテーブル<guitbl>[4]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられ、メニューストリング<menu\_str>[0]～<menu\_str>[3]がGUIテーブル<guitbl>[4]に属する枝メニューテーブル<menu\_tbl>[1]に割り当てられる。

#### 【0056】

“表面処理”については、“無垢”，“ラッカー”，“その他塗装”，“金メッキ”，“銀メッキ”および“その他メッキ”の6つの小項目が枝メニュー“0”に属し、“無垢”，“ラッカー”，“漆”，“ニス”および“その他塗装”の5つの小項目が枝メニュー“1”に属する。このため、メニューストリング<menu\_str>[0]～<menu\_str>[5]がGUIテーブル<guitbl>[5]に属する枝メニューテーブル<menu\_tbl>[0]に割り当てられ、メニューストリング<menu\_str>[0]～<menu\_str>[4]がGUIテーブル<guitbl>[5]に属する枝メニューテーブル<menu

\_\_tbl>[1]に割り当てられる。

#### 【0057】

ツリーテーブルは、図2～図4に示す依存関係毎に用意される。“音楽ジャンル”には枝メニュー“0”が設けられ、枝メニュー“0”の下に1つの依存関係が存在する。このため、GUIテーブル<guitbl>[0]に属する枝メニューテーブル<menu\_\_tbl>[0]にツリーテーブル<tree\_\_tbl>[0]が割り当てられる。“楽器分類”にも枝メニュー“0”が設けられ、枝メニュー“0”の下に1つの依存関係が存在する。このため、GUIテーブル<guitbl>[1]に属する枝メニューテーブル<menu\_\_tbl>[0]にツリーテーブル<tree\_\_tbl>[0]が割り当てられる。“楽器名”には枝メニュー“0”～“4”が設けられ、かかる枝メニュー“0”～“4”の下に依存関係が1つずつ存在する。このため、GUIテーブル<guitbl>[2]に属する枝メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[4]にツリーテーブル<tree\_\_tbl>[0]が1つずつ割り当てられる。

#### 【0058】

“材質”には、枝メニュー“0”～“2”が設けられる。枝メニュー“0”の下には7つの依存関係が存在し、枝メニュー“1”の下には1つの依存関係が存在し、そして枝メニュー“2”の下には6つの依存関係が存在する。このため、GUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_\_tbl>[0]にツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[6]が割り当てられ、GUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_\_tbl>[1]にツリーテーブル<tree\_\_tbl>[0]が割り当てられ、そしてGUIテーブル<guitbl>[3]に属する枝メニューテーブル<menu\_\_tbl>[2]にツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[5]が割り当てられる。

#### 【0059】

“材料”には枝メニュー“0”および“1”が設けられ、かかる枝メニュー“0”および“1”の下に依存関係が2つずつ存在する。このため、GUIテーブル<guitbl>[4]に属する枝メニューテーブル<menu\_\_tbl>[0]にツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[1]が割り当てられ、GUIテーブル<guitbl>[4]に属する枝メニューテーブル<menu\_\_tbl>[1]にツリーテーブル<tree\_\_tbl>[0]～<tre

e\_tbl>[1]が割り当てられる。

#### 【0060】

“表面処理”にも枝メニュー“0”および“1”が設けられ、かかる枝メニュー“0”および“1”の下に依存関係が2つずつ存在する。このため、GUIテーブル<guitbl>[5]に属する枝メニューテーブル<menu\_tbl>[0]にツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[1]が割り当てられ、GUIテーブル<guitbl>[5]に属する枝メニューテーブル<menu\_tbl>[1]にツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[1]が割り当てられる。

#### 【0061】

LANテーブル<lantable>[0]は、\*LAN\_GUI\_TABLEおよびLAN\_GUI\_MAXNUMを含む。\*LAN\_GUI\_TABLEを形成する4バイトには、GUIテーブル<guitbl>[0]～<guitbl>[N-1]の先頭アドレスが記述される。LAN\_GUI\_MAXNUMを形成する1バイトには、“N”の具体的な数値(=6)が記述される。

#### 【0062】

GUIテーブル<guitbl>[0]～<guitbl>[N-1]の各々は、\*GUI\_TABLE, GUI\_SIZE, GUI\_VISIBLE, GUI\_SELECT, GUI\_PROPERTY, \*GUI\_LINKADR, GUI\_MAXNUMおよびGUI\_MEMBERを含む。\*GUI\_TABLEを形成する4バイトには、文字列データGUICONFO.DATに含まれる所望の文字列の先頭アドレスが記述される。所望の文字列は、GUIテーブル<guitbl>[0]にとっては“音楽ジャンル”であり、GUIテーブル<guitbl>[1]にとっては“楽器分類”であり、GUIテーブル<guitbl>[2]にとっては“楽器名”である。また、GUIテーブル<guitbl>[3]にとっては“材質”であり、GUIテーブル<guitbl>[4]にとっては“材料”であり、GUIテーブル<guitbl>[5]にとっては“表面処理”である。

#### 【0063】

GUI\_SIZEを形成する1バイトには、表示する文字列のサイズが記述される。たとえば“音楽ジャンル”のサイズは12バイトであり、“楽器分類”のサイズは8バイトである。

#### 【0064】

GUI\_VISIBLEを形成する1バイトには、表示の可否を示す識別子が記述される

。この実施例では、全ての文字列が表示可能であるため、GUI\_VARIABLEには常に“表示可”を示す識別子が記述される。

#### 【0065】

GUI\_SELECTを形成する1バイトには、選択の可否を示す識別子が記述される。上述の依存関係から、選択が許容される大項目と選択が禁止される大項目とが存在する。したがって、図7(A)に示す大項目メニューが表示されるとき、GUIテーブル<guitbl>[0]～<guitbl>[1]のGUI\_SELECTには“選択可”を示す識別子が記述されるが、GUIテーブル<guitbl>[3]～<guitbl>[5]のGUI\_SELECTには“選択不可”を示す識別子が記述される。

#### 【0066】

GUI\_PROPERTYを形成する1バイトには、自分に属する小項目の選択状態を示す識別子が記述される。自分に属する小項目が未だ選択されていなければ“項目未選択”を示す識別子が記述され、自分に属する小項目が選択済みであれば“項目選択済み”を示す識別子が記述され、そして自分に属する小項目が変更不可能にロックされていれば“項目ロック”を示す識別子が記述される。

#### 【0067】

\*GUI\_LINKADRを形成する4バイトには、自分に属する枝メニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]の先頭アドレスが記述される。また、GUI\_MAXNUMを形成する1バイトには、“I”の具体的な数値(<guitbl>[2]については“5”)が記述される。

#### 【0068】

GUI\_MEMBERを形成する3バイトには、選択項目情報が記述される。つまり、自分に属する小項目が選択済みあるいはロックされているときは、この小項目が属する枝メニューの番号が第1バイトに記述され、この小項目自体の番号が第2バイトに記述される。自分に属する小項目が未選択のときは、“-1”が第1バイトおよび第2バイトに記述される。なお、第0バイトは予約領域である。

#### 【0069】

枝メニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]の各々は、\*str\_table, str\_maxnum, tree\_maxnumおよび\*tree\_tableを含む。\*str\_tableを形成

する 4 バイトには、自分に属するメニューストリング<menu\_str>[0]～<menu\_str>[K-1]の先頭アドレスが記述される。str\_maxnumを形成する 1 バイトには、“K”の具体的な数値 (<guitbl>[2]に属する<menu\_tbl>[0]については“4”) が記述される。\*tree\_tableを形成する 4 バイトには、自分に属するツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[L-1]の先頭アドレスが記述される。tree\_maxnumを形成する 1 バイトには、“L”の具体的な数値 (<guitbl>[3]に属する<menu\_tbl>[2]については“6”) が記述される。

#### 【0070】

メニューストリング<menu\_str>[0]～<menu\_str>[K-1]の各々は、\*m\_string, m\_length, およびm\_freeを含む。\*m\_stringを形成する 4 バイトには、文字列データ GUICONF0.DATに含まれる所望の文字列の先頭アドレスが記述される。GUI テーブル<guitbl>[0]に属する枝メニューテーブル<menu\_tbl>[0]の下メニューストリング<menu\_str>[0]～<menu\_str>[K-1]に注目した場合、所望の文字列は、メニューストリング<menu\_str>[0]にとっては“クラシック”であり、メニューストリング<menu\_str>[1]にとっては“ジャズ”であり、メニューストリング<menu\_str>[2]にとっては“その他”である。

#### 【0071】

m\_lengthを形成する 1 バイトには、所望の文字列のサイズが記述される。m\_freeを形成する 1 バイトには、表示の可否を示す識別子が記述される。この実施例では、\*m\_stringが示す文字列つまり小項目は全て表示される。このため、m\_freeには“表示可”を示す識別子が常に記述される。

#### 【0072】

ツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[L-1]の各々は、gui\_treeを含む。gui\_treeを形成する 3 バイトには、依存関係情報が記述される。つまり、第 0 バイトには依存する大項目の番号が記述され、第 1 バイトには依存する枝メニューの番号が記述され、そして第 2 バイトには依存する小項目の番号が記述される。たとえば、GUI テーブル<guitbl>[2]に属する枝メニューテーブル<menu\_tbl>[2]の下ツリーテーブル<tree\_tbl>[0]に注目した場合、gui\_treeの第 0 バイト、第 1 バイトおよび第 2 バイトには、“1”，“0” および“2” がそ

れぞれ記述される。

### 【0073】

図17を参照して、文字列データGUICONF0.DATは、メニュー画面に表示される文字列のデータを含む。各々の文字列を形成する1文字（全角文字）は2バイトを用いて記述される。また、各々の文字列は、ヌルコードが記述された1バイトによって区切られる。たとえば、“音楽ジャンル”は12バイトを用いて記述され、“楽器分類”は8バイトを用いて記述され、そして“音楽ジャンル”と“楽器分類”との間にヌルコードが記述される。

### 【0074】

続いて、メニューファイルデータをSDRAM22に確保する処理、ならびに表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATに基づくメニュー処理に重点を置いて、CPU44の詳しい処理を説明する。CPU44は、フラッシュメモリ42に格納されたかつ図20～図42に示すフロー図に対応する制御プログラムを実行する。

### 【0075】

まず、図20に示すステップS1で、スロット36に装着されたカードがメモ리카ード38および通信カード40のいずれであるかを判別する。装着されたカードがメモ리카ード38であればステップS7に進み、このメモ리카ード38にメニュー設定ファイルが記録されているかどうか判断する。NOと判断されたときはステップS27に進むが、YESと判断されたときはステップS9に進む。

### 【0076】

ステップS9では、このメニュー設定ファイルに格納された表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATをメモ리카ード38からSDRAM22に転送する。この転送処理によって、表示制御テーブルGUICONF0.TBLのサイズデータが図18に示すSDRAM22のアドレス0x49000000から始まる4バイトに書き込まれ、この4バイトに続いて表示制御テーブルGUICONF0.TBLが書き込まれる。この実施例では、表示制御テーブルGUICONF0.TBLは0x34Cバイトのサイズを有し、表示制御テーブルGUICONF0.TBLの末尾はアドレス0x4900034Fに書き込まれる。アドレス0x49000350から始まる4バイトには文字列データGUICONF0.D

ATのサイズデータが書き込まれ、この4バイトに続いて文字列データGUICONF0.DATが書き込まれる。文字列データGUICONF0.DATは0x1E9バイトのサイズを有し、文字列データGUICONF0.DATの末尾はアドレス0x4900053Dに書き込まれる。

#### 【0077】

ステップS11ではSDRAM22に転送された表示制御テーブルGUICONF0.TBLをフラッシュメモリ42に格納された表示制御テーブルGUICONF0.TBLと比較し、ステップS13では両者が一致するかどうかを比較する。両者が不一致の場合、あるいはフラッシュメモリ42に表示制御テーブルGUICONF0.TBLが存在しない場合は、ステップS13でNOと判断し、ステップS19で表示制御テーブルGUICONF0.TBLおよびそのサイズデータをSDRAM22上で移動させる。具体的には、図18に示すアドレス0x49000540以降への複写を行う。

#### 【0078】

ステップS21では表示制御テーブルGUICONF0.TBLが有するアドレス値をオフセット処理によって補正し、ステップS23では補正された表示制御テーブルGUICONF0.TBLが有効であるかどうかを判断する。ここでNOと判断されたときはステップS27に進むが、YESと判断されたときはステップS25に進む。ステップS25では、SDRAM22に格納されたオリジナルの表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATと補正された表示制御テーブルGUICONF0.TBLとをこれらのサイズデータとともにフラッシュメモリ42に転送する。オリジナルの表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATと補正された表示制御テーブルGUICONF0.TBLとこれらのサイズデータとは、図19に示す要領でフラッシュメモリ42に格納される。ステップS25の処理が完了すると、ステップS47に進む。

#### 【0079】

ステップS13でYESと判断されたときはステップS15に進み、補正された表示制御テーブルGUICONF0.TBLとそのサイズデータとをフラッシュメモリ42からSDRAM22に転送する。補正された表示制御テーブルGUICONF0.TBLとそのサイズデータとは、図18に示すアドレス0x49000540以降に書き込まれる。ステップS17では、ステップS23と同様の判断処理を行い、NOと判断された



ときはステップS27に進み、YESと判断されたときはステップS47に進む。

#### 【0080】

スロット36に通信カード40が装着されているときは、ステップS3およびS5でステップS15およびS17と同様の処理を実行する。そして、ステップS5でNOと判断されたときはステップS67に進み、ステップS5でYESと判断されたときはステップS97に進む。

#### 【0081】

図21に示すステップS27ではメモ리카ード38の空き容量を取得し、続くステップS29では取得した空き容量を閾値MINと比較する。空き容量が閾値MIN以下のときは、ステップS29でNOと判断し、ステップS31で警告を発生する。これに対して、空き容量が閾値MINを上回るときは、ステップS29からステップS33に進み、次の撮影によって得られるJPEGデータを格納する画像ファイルのファイル名を決定する。

#### 【0082】

ステップS35ではシャッターキー46の操作の有無を判別し、ステップS37ではメニューキー48の操作の有無を判別する。メニューキー48が操作されたときはステップS39で特別メニュー処理を行い、処理が完了するとステップS39に戻る。特別メニュー処理によって、図7(A)～図14(B)に示す要領で詳細情報が作成される。シャッターキー46が操作されたときは、ステップS41で撮影処理を行い、ステップS43でファイルヘッダを作成する。ステップS41の撮影処理によって、撮影画像のJPEGデータが得られる。また、ステップS43の処理によって、特別メニュー処理で作成された詳細情報がファイルヘッダに埋め込まれる。ステップS45では、JPEGデータおよびファイルヘッダを含む画像ファイルをメモ리카ード38に記録する。ステップS45の処理が完了すると、ステップS27に戻る。

#### 【0083】

図22に示すステップS47～S65の処理は、ステップS59で通常メニュー処理を行い、通常メニュー処理によって作成されたメニュー情報を含むファイ

ルヘッダをステップS 67で作成する点を除き、上述のステップS 27～S 45と同様であるため、重複した説明は省略する。

#### 【0084】

図23に示すステップS 67では、サーバとの接続状態が確保されているかどうか判断する。ここでNOと判断されると、ステップS 75でフラグlan\_flgをリセットし、フラッシュメモリ42に十分な空き容量があるかどうかをステップS 77で判断する。空き容量が十分であるときはステップS 77からステップS 79に進むが、空き容量が不十分であるときはステップS 75からステップS 67に戻る。

#### 【0085】

ステップS 67でYESと判断されると、ステップS 69でフラグlan\_flgをセットし、フラッシュメモリ42に未送信の画像ファイルが存在するかどうかをステップS 71で判断する。NOと判断されたときはステップS 79に進み、YESと判断されたときはステップS 73のサーバ転送処理を経てステップS 67に戻る。サーバ転送処理によって、未送信画像ファイルがフラッシュメモリ42から読み出され、読み出された未送信画像ファイルが通信カード40を経てサーバに転送される。

#### 【0086】

ステップS 79～S 89では、上述のステップS 33～S 43と同様の処理を行う。したがって、メニューキー48の操作によって所望の詳細情報が作成され、シャッターキー46の操作によって撮影画像のJPEGデータと詳細情報とを含む画像ファイルが作成される。ステップS 91ではフラグlan\_flgの状態を判別する。フラグlan\_flgがセット状態のときはステップS 93に進み、作成された画像ファイルを通信カード40を介してサーバに転送する。フラグlan\_flgがリセット状態のときはステップS 95に進み、作成された画像ファイルをフラッシュメモリ42に記録する。ステップS 93またはS 95の処理が完了すると、ステップS 67に戻る。

#### 【0087】

図24を参照して、ステップS 97ではサーバとの接続状態が確保されている

かどうかを判断する。接続状態が確保されていなければステップS9に進み、タイムアウトが発生したかどうかを判断する。そして、タイムアウトが発生していなければステップS97に戻るが、タイムアウトが発生するとステップS105で警告が発生する。

**【0088】**

サーバとの接続状態が確保されると、ステップS101でサーバの空き容量を取得し、ステップS103でこの空き容量を閾値MINと比較する。空き容量が閾値MIN以下であればステップS105で警告が発生するが、空き容量が閾値MINを上回ればステップS107～S117で上述のステップS55～S63と同様の処理を行う。したがって、メニューキー48の操作によって所望のメニュー情報が作成され、シャッターキー46の操作によって撮影画像のJPEGデータとメニュー情報とを含む画像ファイルが作成される。ステップS119ではステップS93と同様のサーバ転送処理を行い、処理が完了するとステップS97に戻る。

**【0089】**

図20に示すステップS21のオフセット処理は、図25～図29に示すサブルーチンに従う。まず図25を参照して、ステップS201では数1に従ってオフセット値TBL\_Offsetを算出し、ステップS203では数2に従ってオフセット値DAT\_Offsetを算出し、そしてステップS205では数3に従ってアドレス値adrを算出する。

**【0090】****【数1】**

TBL\_Offset＝

$(0x49000000 + \text{表示制御テーブルサイズ} + \text{文字列データサイズ} + 8) + 4$

**【0091】****【数2】**

DAT\_Offset＝ $(0x001E0000 + \text{表示制御テーブルサイズ} + 4) + 4$

**【0092】****【数3】**

$\text{adr} = 0x49000000 + \text{表示制御テーブルサイズ} + \text{文字列データサイズ} + 8$

数1の演算によって、図18に示すアドレス値0x49000000と、オリジナルの表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATのサイズ値と、各々のサイズデータを記述するためのサイズ値(=8バイト)と、補正された表示制御テーブルGUICONF0.TBLのサイズデータを記述するためのサイズ値(=4バイト)とが、互いに加算される。TBL\_Offsetは、補正された表示制御テーブルGUICONF0.TBLの先頭アドレスを指向する。

#### 【0093】

数2の演算によって、図19に示すアドレス値0x001E0000と、オリジナルの表示制御テーブルGUICONF0.TBLのサイズ値と、そのサイズデータを記述するためのサイズ値(=4バイト)と、オリジナルの文字列データGUICONF0.DATのサイズデータを記述するためのサイズ値(=4バイト)とが、互いに加算される。DAT\_Offsetは、オリジナルの文字列データGUICONF0.DATの先頭アドレスを指向する。

#### 【0094】

数3の演算によって、図18に示すアドレス値0x49000000と、オリジナルの表示制御テーブルGUICONF0.TBLおよび文字列データGUICONF0.DATのサイズ値と、各々のサイズデータを記述するためのサイズ値(=8バイト)とが、互いに加算される。変数adrは、補正された表示制御テーブルGUICONF0.TBLのデータサイズが記述された4バイトの先頭アドレスを指向する。

#### 【0095】

ステップS207では、変数sizeとして変数adr以降の4バイト値を設定する。変数sizeは、補正された表示制御テーブルGUICONF0.TBLのサイズを示すこととなる。ステップS209では、変数adrと“4”との加算値を変数lantableとして設定する。これによって、図16に示すLANテーブル<lantable>[0]が注目される。

#### 【0096】

ステップS211では、LANテーブル<lantable>[0]に含まれる\*LAN\_GUI\_TABLEの最上位ビット値を判別する。最上位ビット値が“0”であればステップS213に進み、最上位ビット値が“1”であればステップS215に進む。ス

ステップS 2 1 3では、\*GUI\_TABLEに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*LAN\_GUI\_TABLEに記述する。ステップS 2 1 5では、\*LAN\_GUI\_TABLEに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセット値DAT\_Offsetを加算し、そして加算値を\*LAN\_GUI\_TABLEに記述する。この実施例では、\*LAN\_GUI\_TABLEの最上位ビット値は常に“0”であり、必ずステップS 2 1 3の処理が実行される。

#### 【0097】

ステップS 2 1 7では、ステップS 2 1 3またはS 2 1 5で更新されたアドレス値を変数guitblとして設定する。これによって、図16に示すGUIテーブル<guitbl>[0]が注目される。ステップS 2 1 9では変数iを“0”に設定し、ステップS 2 2 1ではGUIテーブル<guitbl>[i]に含まれる\*GUI\_TABLEの最上位ビット値を判別する。

#### 【0098】

最上位ビット値が“0”であればステップS 2 2 3に進み、最上位ビット値が“1”であればステップS 2 2 5に進む。ステップS 2 2 3では、\*GUI\_TABLEに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*GUI\_TABLEに記述する。ステップS 2 2 5では、\*GUI\_TABLEに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセット値DAT\_Offsetを加算し、そして加算値を\*GUI\_TABLEに記述する。この実施例では、\*GUI\_TABLEの最上位ビット値は常に“1”であり、必ずステップS 2 2 5の処理が実行される。

#### 【0099】

ステップS 2 2 3またはS 2 2 5の処理が完了するとステップS 2 2 7に進み、GUIテーブル<guitbl>[i]に含まれる\*GUI\_LINKADRの最上位ビット値を判別する。最上位ビット値が“0”であればステップS 2 2 9に進み、最上位ビット値が“1”であればステップS 2 3 1に進む。ステップS 2 2 9では、\*GUI\_LINKADRに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*GUI\_LINKADRに記述する。ステップS 2 3 1では、\*GUI\_LINKADRに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセッ

ト値DAT\_Offsetを加算し、そして加算値を\*GUI\_LINKADRに記述する。この実施例では、\*GUI\_LINKADRの最上位ビット値は常に“0”であり、必ずステップS229の処理が実行される。

#### 【0100】

ステップS233では変数iをインクリメントし、ステップS235では更新された変数iをLANテーブル<lantable>[0]の\*LAN\_GUI\_MAXNUMが示す数値Nと比較する。そして、変数iが数値Nを下回る限り、ステップS221～S233の処理を繰り返す。変数iが数値Nに達すると、ステップS235からステップS237に進む。

#### 【0101】

ステップS237では変数iを再度“0”に設定し、ステップS239ではGUIテーブル<guitbl>[i]に含まれる\*GUI\_LINKADRが示すアドレス値（更新されたアドレス値）を変数menu\_tblとして設定し、ステップS241では変数jを“0”に設定する。ステップS243では、GUIテーブル<guitbl>[i]に属する枝メニューテーブル<menu\_tbl>[j]について、\*str\_tableの最上位ビット値を判別する。変数iおよびjがいずれも“0”のときは、GUIテーブル<guitbl>[0]に属する枝メニューテーブル<menu\_tbl>[0]について、\*str\_tableの最上位ビット値が判別される。

#### 【0102】

この最上位ビット値が“0”であればステップS245に進み、最上位ビット値が“1”であればステップS247に進む。ステップS245では、\*str\_tableに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*str\_tableに記述する。ステップS247では、\*str\_tableに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセット値DAT\_Offsetを加算し、そして加算値を\*str\_tableに記述する。この実施例では、\*str\_tableの最上位ビット値は常に“0”であり、必ずステップS245の処理が実行される。

#### 【0103】

ステップS249では注目する枝メニューテーブル<menu\_tbl>[j]に含まれる\*

tree\_tableの最上位ビット値を判別する。この最上位ビット値が“0”であればステップS 2 5 1に進み、最上位ビット値が“1”であればステップS 2 5 3に進む。ステップS 2 5 1では、\*tree\_tableに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*tree\_tableに記述する。ステップS 2 5 3では、\*tree\_tableに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセット値DAT\_Offsetを加算し、そして加算値を\*tree\_tableに記述する。この実施例では、\*tree\_tableの最上位ビット値は常に“0”であり、必ずステップS 2 5 1の処理が実行される。

#### 【0104】

ステップS 2 5 1またはS 2 5 3の処理が完了するとステップS 2 5 5に進み、直前のステップS 2 4 5またはS 2 4 7で更新されたアドレス値を変数menu\_strとして設定する。これによって、注目する枝メニューテーブル<menu\_tbl>[j]の下のメニューストリング<menu\_str>[0]～<menu\_str>[K-1]が注目される。

#### 【0105】

ステップS 2 5 7では変数kを“0”に設定し、ステップS 2 5 9ではメニューストリング<menu\_str>[k]に含まれる\*m\_stringの最上位ビット値を判別する。変数i, jおよびkがいずれも“0”であれば、GUIテーブル<guitbl>[0]に属する枝メニューテーブル<menu\_tbl>[0]の下のメニューストリング<menu\_str>[0]について、\*m\_stringの最上位ビット値が判別される。

#### 【0106】

この最上位ビット値が“0”であればステップS 2 6 1に進み、最上位ビット値が“1”であればステップS 2 6 3に進む。ステップS 2 6 1では、\*m\_stringに記述されたアドレス値にオフセット値TBL\_Offsetを加算し、加算値を\*m\_stringに記述する。ステップS 2 6 3では、\*m\_stringに記述されたアドレス値の最上位ビット値を“0”に変更し、変更されたアドレス値にオフセット値DAT\_Offsetを加算し、そして加算値を\*m\_stringに記述する。この実施例では、\*m\_stringの最上位ビット値は常に“1”であり、必ずステップS 2 6 3の処理が実行される。

#### 【0107】

ステップS261またはS263の処理が完了すると、ステップS265で変数kをインクリメントする。ステップS267では、更新された変数kを枝メニューテーブル<menu\_tbl>[j]のstr\_maxnumが示す数値Kと比較する。そして、変数kが数値Kを下回る限り、ステップS259～S265の処理を繰り返す。変数kが数値Kに達すると、ステップS267でNOと判断し、ステップS269で変数jを再度インクリメントする。

#### 【0108】

ステップS271では、更新された変数jをGUIテーブル<guitbl>[j]のGUI\_MAXNUMが示す数値Iと比較する。そして、変数jが数値Iを下回る限りステップS243～S269の処理を繰り返すが、変数jが数値Iに達すると、ステップS273で変数iをインクリメントしてステップS275に進む。ステップS275では、更新された変数iをLANテーブル<lantable>[0]の\*LAN\_GUI\_MAXNUMが示す数値Nと比較する。そして、変数iが数値Nを下回る限りステップS239～S273の処理を繰り返すが、変数iが数値Nに達すると、上階層のルーチンに復帰する。

#### 【0109】

このようなオフセット処理によって、図18に示すアドレス0x49000540以降に複写された表示制御テーブルGUICONF0.TBL内でのリンク、ならびにこの表示制御テーブルGUICONF0.TBLと図19に示す文字列データGUICONF0.TBLとの間のリンクが確保される。

#### 【0110】

図21に示すステップS39または図23に示すステップS85の特別メニュー処理は、図30～図42に示すサブルーチンに従う。

#### 【0111】

まず図30に示すステップS301で各種の変数を初期化する。具体的には、変数cnt0を“0”に設定し、変数cnt1を“1”に設定し、変数pre\_cnt0を“0”に設定し、変数pre\_cnt1を“0”に設定し、そして変数modeを“1”に設定する。ここで、変数cnt0は、大項目メニューおよび枝メニューのいずれを画面に表示すべきかを識別するための変数である。cnt0=0は大項目メニューの表示を



意味し、cnt0=1は枝メニューの表示を意味する。変数cnt1はカーソルCSの表示位置を識別するための変数である。変数cnt1の数値が大きくなるほど、カーソルCSの表示位置は下方に移動する。変数pre\_cnt0およびpre\_cnt1はそれぞれ、変数cnt0およびcnt1の前の数値である。変数modeは、メニュー画面の描画／クリアを識別するための変数である。mode=1はメニュー画面の描画を意味し、mode=-1はメニュー画面のクリアを意味する。

#### 【0112】

ステップS303では、メニュー表示処理を実行する。これによって、図7(A)に示す大項目メニューが画面に表示される。メニュー表示が完了すると、ステップS305で変数tree\_numの値を判別する。変数tree\_numはカーソルCSが指向する項目の番号を示す変数である。ただし、メニュー表示処理にエラーが生じると、変数tree\_numは“-1”を示す。

#### 【0113】

変数tree\_numが“-1”のときは、ステップS305からステップS307に進み、変数cnt1を閾値MAX\_NUM-1と比較する。ここで、MAX\_NUMは表示可能な項目数の上限値である。変数cnt1が閾値MAX\_NUM-1に満たなければ、ステップS309で変数cnt1をインクリメントしてからステップS303に戻る。変数cnt1が閾値MAX\_NUM-1に達すると、ステップS307でNOと判断してエラー処理を実行する。

#### 【0114】

変数tree\_numが“-1”以外のときは、ステップS305でYESと判断し、ステップS311～S317でキー操作を判別する。具体的には、メニューキー48の操作の有無をステップS311で判別し、アップキー52の操作の有無をステップS313で判別し、ダウンキー54の操作の有無をステップS315で判別し、そしてセットキー50の操作の有無をステップS317で判別する。

#### 【0115】

メニューキー48が操作されると、ステップS311から図31に示すステップS319に進み、変数cnt0の値を判別する。変数cnt0が“0”であれば、つまり画面に大項目メニューが表示されていれば、メニュー画面をクリアするべく、

ステップS 3 2 1以降の処理を実行する。ステップS 3 2 1では変数cnt0, cnt1およびmodeをそれぞれ“0”, “0”および“-1”に設定し、ステップS 3 2 3ではメニュー表示処理を実行する。メニュー表示処理によるメニュー画面のクリアが完了すると、上階層のルーチンに復帰する。

#### 【0116】

これに対して、変数cnt0が“1”であれば、つまり画面に枝メニューが表示されていれば、枝メニューから大項目メニューに復帰するべく、ステップS 3 2 5以降の処理を行う。ステップS 3 2 5では、変数cnt0を“0”に戻し、変数cnt1を変数back\_cnt1に設定し、そして変数modeを“1”に設定する。変数back\_cnt1は、大項目メニューから枝メニューへ遷移するときに変数cnt1の値を一時的に退避させるための変数である。ステップS 3 2 5の処理によって、退避されていた数値が変数cnt1に復帰する。ステップS 3 2 7ではメニュー表示処理を実行し、これによって大項目メニューが画面に表示される。ステップS 3 2 7の処理が完了すると、ステップS 3 1 1に戻る。

#### 【0117】

アップキー52が操作されると、図30に示すステップS 3 1 3から図32に示すステップS 3 2 9に進み、変数loopを“0”に設定する。変数loopは、カーソルCSの表示位置をリング状に移動させるべきかどうかを示す変数である。“0”はリング状の移動が不要であることを示し、“1”はリング状の移動が必要であることを示す。ステップS 3 3 1では、変数cnt1の数値を判別する。変数cnt1が“0”よりも大きければ、カーソルCSは上方に移動可能であるとみなし、ステップS 3 3 7で変数cnt1をデクリメントする。ステップS 3 3 9では変数modeを“1”に設定し、ステップS 3 4 1ではカーソルCSが上方に移動させるべくメニュー表示処理を行う。ステップS 3 4 3では、変数tree\_numの数値を判別する。変数tree\_numが“-1”であればメニュー表示処理にエラーが生じたとみなしてステップS 3 3 1に戻り、変数tree\_numが“0”以上であればメニュー表示処理が適切に実行されたとみなしてステップS 3 1 1に戻る。

#### 【0118】

ステップS 3 3 1でNOと判断されたときは、ステップS 3 3 3で変数loopの

値を判別する。そして、変数loopが“1”であればエラー処理を実行するが、変数loopが“0”であれば、ステップS335で変数loopを“1”に更新し、かつ変数cnt1を閾値MAX\_NUM-1に設定する。ステップS335の処理が完了するとステップS341に進み、カーソルCSを画面の最下欄に移動させるべく、メニュー表示処理を行う。

#### 【0119】

ダウンキー54が操作されたときは、図30に示すステップS315でYESと判断し、図33に示すステップS345～S359の処理を実行する。ただし、この一連の処理は、ステップS347で変数cnt1を閾値MAX\_NUM-1と比較する点、ステップS353で変数cnt1をインクリメントする点、ならびにステップS351で変数cnt1を“0”に設定する点を除き、上述のステップS329～S343と同様であるため、重複した説明は省略する。

#### 【0120】

セットキー50が操作されたときは、図30に示すステップS317でYESと判断し、図34に示すステップS361に進む。ステップS361では、大項目メニューおよび枝メニューのいずれが画面に表示されているかを変数cnt0に基づいて判別する。変数cnt1が“0”であれば、大項目メニューが表示されているとみなし、ステップS363以降で大項目メニューを枝メニューに更新する処理を行う。

#### 【0121】

まずステップS363で、GUIテーブル<guitbl>[cnt1]に含まれるGUI\_PROPERTYの識別子を判別する。この識別子が“項目ロック”を示していれば、そのままステップS311に戻る。この結果、今回のセットキー50の操作は無効とされ、大項目メニューが表示され続ける。識別子が“項目未選択”または“項目選択済み”を示していれば、ステップS365に進み、枝メニューの表示を意味する“1”を変数cnt0に設定し、変数cnt1が示す数値を変数back\_cnt1に退避させる。

#### 【0122】

ステップS367では、GUIテーブル<guitbl>[back\_cnt1]に含まれるGUI

\_\_MEMBERの第2バイト値を参照して、これから表示する枝メニューに選択済みの小項目が存在するかどうかを判別する。この第2バイト値が“0”以上であれば、選択済みの小項目が存在するとみなし、ステップS371でこの第2バイト値を変数cnt1に設定する。これに対して、第2バイト値が“-1”であれば、選択済みの小項目は存在しないとみなして、ステップS369で変数cnt1を“0”に設定する。ステップS373では変数modeを“1”に設定し、ステップS375ではメニュー表示処理を行う。

#### 【0123】

したがって、図7(B)に示すようにカーソルCSを“楽器分類”に合わせた状態でセットキー50を操作すると、画面表示は図8(B)に遷移する。つまり、“楽器分類”に属する小項目はいずれも未選択であるため、カーソルCSは先頭の“弦楽器”に合わせられる。これに対して、図9(A)に示す“木管楽器”にカーソルCSを合わせた状態でセットキー50を操作すると、画面表示は図8(B)に遷移する。つまり、“楽器分類”に属する小項目である“木管楽器”が選択済みであるため、カーソルCSは、“木管楽器”に合わせられる。ステップS375の処理が完了すると、ステップS311に戻る。

#### 【0124】

ステップS361でNOと判断されると、画面には枝メニューが表示されているとみなし、図35に示すステップS377以降で大項目メニューに復帰する処理を行う。まずステップS377で、GUIテーブル<guitbl>[back\_cnt1]に含まれるGUI\_PROPERTYに“項目選択済み”を示す識別子を記述する。続くステップS379では、GUIテーブル<guitbl>[back\_cnt1]に含まれるGUI\_MEMBERの第2バイト値が変数cnt1に等しいかどうかを判別する。

#### 【0125】

GUI\_MEMBERの第2バイト値と変数cnt1とが互いに一致するときは、選択済みの小項目が再度選択されたとみなし、ステップS379からステップS381に進む。ステップS381では、変数back\_cnt1を変数cnt1に設定し、変数cnt0を“0”に設定する。ステップS383では変数modeを“1”に設定する。ステップS383の処理が完了すると、ステップS385でメニュー表示処理を行う。

## 【0126】

たとえば、図9（A）に示す大項目メニューから図8（B）に示す枝メニューに遷移して“木管楽器”を再度選択した場合、GUI\_MEMBERの第2バイト値は変数cnt1と一致する。このとき、ステップS381～S385の処理が実行される。画面表示は、図8（B）に示す枝メニューから図9（A）に示す大項目メニューに戻る。

## 【0127】

GUI\_MEMBERの第2バイト値と変数cnt1とが互いに不一致であれば、未選択の小項目が選択されたとみなして、ステップS379からステップS387に進む。ステップS387では、GUIテーブル<guitbl>[back\_cnt1]に含まれるGUI\_MEMBERの第1バイトおよび第2バイトに、変数tree\_numおよびcnt1をそれぞれ設定する。たとえば、図9（A）に示す大項目メニューから図9（B）に示す枝メニューに遷移して“金管楽器”を選択した場合、GUIテーブル<guitbl>[1]に含まれるGUI\_MEMBERの第1バイトおよび第2バイトに枝メニュー番号“0”および小項目番号“2”が記述される。

## 【0128】

ただし、未選択の小項目を選択することによって、大項目メニュー上に表示すべき項目数が変化する可能性がある。たとえば、図9（B）に示す大項目メニューから図10（A）に示す枝メニューに遷移して“フルート”を選択すると、図11（A）に示すように大項目メニューの項目数が増加する。また、図11（A）に示す大項目メニューから図8（B）に示す枝メニューに遷移し、“金管楽器”を選択すると、図9（A）に示すように（ただし、“木管楽器”に代えて“金管楽器”が表示）大項目メニューの項目数が減少する。このため、ステップS391以降で依存関係を検証する処理を実行する。

## 【0129】

まずステップS389で変数iを“0”に設定し、ステップS391で変数iをLANテーブル<lantable>[0]に含まれるLAN\_GUI\_MAXNUMが示す数値Nと比較する。そして、変数iが数値Nを下回る限りステップS393～S425の処理を繰り返し、変数iが数値Nに達すると、ステップS381に進む。

## 【0130】

ステップS393では、GUIテーブル<guitbl>[i]の\*GUI\_LINKADRに記述されたアドレス値を変数menu\_tblとして設定する。これによって、GUIテーブル<guitbl>[i]に属するメニューテーブル<menu\_tbl>[0]～<menu\_tbl>[K-1]が注目される。

## 【0131】

ステップS395では変数jを“0”に設定し、ステップS397では変数jをGUIテーブル<guitbl>[i]に含まれるGUI\_MAXNUMが示す数値Iと比較し、そしてステップS401ではメニューテーブル<menu\_tbl>[j]に含まれるtree\_maxnumが示す数値Lを判別する。

## 【0132】

変数jが数値I以上のときは、ステップS397からステップS399に進む。ステップS399では、GUIテーブル<guitbl>[i]に含まれるGUI\_SELECTに“選択不可”の識別子を記述し、GUIテーブル<guitbl>[i]に含まれるGUI\_PROPERTYに“項目未選択”の識別子を記述し、そしてGUIテーブル<guitbl>[i]に含まれるGUI\_MEMBERの第1バイトおよび第2バイトに“－1”を記述する。ステップS399の処理が完了すると、ステップS403で変数iをインクリメントしてからステップS391に戻る。なお、ステップS399の処理は、後段のステップS423の処理と合わせて説明する。

## 【0133】

変数jが数値I未満でかつ数値Lが“0”を示していれば、GUIテーブル<guitbl>[i]の下にはメニューテーブル<menu\_tbl>[j]が存在するものの、このメニューテーブル<menu\_tbl>[j]の下にはツリーテーブルは存在しないとみなし、ステップS401でYESと判断する。そして、ステップS403で変数iをインクリメントしてステップS391に戻る。たとえば、図9(A)に示す“音楽ジャンル”にはいずれのツリーテーブルも存在しないため、GUIテーブル<guitbl>[0]に注目したときは、ステップS401でYESと判断される。

## 【0134】

変数jが数値I未満でかつ数値Lが“1”以上であれば、ステップS401から

ステップS 4 0 5に進み、メニューテーブル<menu\_\_tbl>[j]に含まれる\*tree\_\_tableのアドレス値を変数tree\_\_tblに設定する。これによって、G U I テーブル<guitbl>[i]の下のメニューテーブル<menu\_\_tbl>[j]に属する<tree\_\_tbl>[0]～<tree\_\_tbl>[L-1]が注目される。

#### 【0 1 3 5】

ステップS 4 0 7では変数kを“0”に設定し、ステップS 4 0 9では変数kをメニューテーブル<menu\_\_tbl>[j]に含まれるtree\_\_maxnumが示す数値Lと比較する。そして、変数kが数値Lに達したときはステップS 4 1 1で変数jをインクリメントしてステップS 3 9 7に戻るが、変数kが数値L未満であればステップS 4 1 3以降の処理を実行する。

#### 【0 1 3 6】

ステップS 4 1 3およびS 4 1 5では、メニューテーブル<menu\_\_tbl>[j]に属するツリーテーブルtree\_\_tbl[k]とこのツリーテーブルtree\_\_tbl[k]を管理するG U I テーブル<guitbl>[tree\_\_tbl[k]. gui\_\_tree[0]]とに注目し、ツリーテーブルtree\_\_tbl[k]のgui\_\_treeとG U I テーブル<guitbl>[tree\_\_tbl[k]. gui\_\_tree[0]]のGUI\_MEMBERとの間で比較を行う。具体的には、gui\_\_treeの第1バイト値とGUI\_MEMBERの第1バイト値とをステップS 4 1 3で比較し、gui\_\_treeの第2バイト値とGUI\_MEMBERの第2バイト値とをステップS 4 1 5で比較する。

#### 【0 1 3 7】

そして、第1バイト値および第2バイト値の少なくとも一方が互いに不一致であれば、ツリーテーブルtree\_\_tbl[k]はG U I テーブル<guitbl>[tree\_\_tbl[k]. gui\_\_tree[0]]に依存しているとみなし、ステップS 4 1 7で変数kをインクリメントしてステップS 4 0 9に戻る。これに対して、第1バイト値および第2バイト値の両方が互いに一致すれば、ツリーテーブルtree\_\_tbl[k]はG U I テーブル<guitbl>[tree\_\_tbl[k]. gui\_\_tree[0]]に依存しているとみなし、ステップS 4 1 9に進む。たとえば、図9 (A)に示す大項目メニューでは、“木管楽器”の選択によってG U I テーブル<guitbl>[1]のGUI\_MEMBERに枝メニュー番号“0”および小項目番号“1”が記述される。このため、G U I テーブル<guitbl>[2]に属するツリーテーブル<tree\_\_tbl>[1]に注目したときに、ステップS 4 1 9

に進む。

#### 【0138】

ステップS419では、GUIテーブル<guitbl>[i]に含まれるGUI\_SELECTに“選択可”を示す識別子を記述する。ここで、GUIテーブル<guitbl>[i]はGUIテーブル<guitbl>[tree\_tbl[k]. gui\_tree[0]]と同じテーブルであり、“選択可”を示す識別子は、ツリーテーブルtree\_tbl[k]が依存するGUIテーブルのGUI\_SELECTに記述される。

#### 【0139】

ステップS421では、ツリーテーブル<tree\_tbl>[k]に含まれるgui\_treeの第0バイト値が変数back\_cnt1に等しいかどうか判断する。そして、YESであれば、ステップS423に進み、GUIテーブル<guitbl>[i]に含まれるGUI\_MEMBERの第1バイトおよび第2バイトに“-1”を設定するとともに、GUIテーブル<guitbl>[i]に含まれるGUI\_PROPERTYに“項目未選択”の識別子を記述する。ステップS423の処理が完了すると、ステップS425で変数iをインクリメントしてからステップS391に戻る。これに対して、ステップS421でNOと判断されると、ステップS423を経ることなくステップS425で変数iをインクリメントし、その後ステップS391に戻る。

#### 【0140】

ステップS399およびS423が処理されるのは、たとえば図9(B)に示す大項目メニューの“木管楽器”を“金管楽器”に変更したときである。つまり、図9(B)に示す大項目メニューから図8(B)に示す枝メニューに遷移し、“金管楽器”にカーソルCSを合わせてセットキー50を操作すると、ステップS399およびS423の処理が実行される。具体的には、“材質”，“材料”および“表面処理”の文字列を非表示とするべくステップS399の処理が実行され、“フルート”を“楽器名”に変更するべくステップS423の処理が実行される。なお、“木管楽器”の“金管楽器”への変更処理は、ステップS387によって行われる。

#### 【0141】

ステップS303, S323, S327, S341, S357, S375また



は S 3 8 5 におけるメニュー表示処理は、図 3 8 ～ 図 4 3 に示すサブルーチンに従う。

#### 【0142】

図 3 8 を参照して、ステップ S 5 0 1 では変数 `tree_num` を “0” に設定し、ステップ S 5 0 3 では変数 `mode` が示す数値を判別する。変数 `mode` が “-1” であれば、つまり変数 `mode` がメニュー画面のクリアを意味していれば、ステップ S 5 0 3 で YES と判断し、ステップ S 5 0 5 で変数 `pre_cnt1` および `pre_cnt2` を “0” に設定するとともに、ステップ S 5 0 7 でメニュー画面をクリアする。ステップ S 5 0 7 の処理が完了すると、上階層のルーチンに復帰する。変数 `mode` が “1” であれば、つまり変数 `mode` がメニュー画面の描画を意味していれば、ステップ S 5 0 3 からステップ S 5 0 9 に進み、変数 `cnt0` が示す数値を判別する。変数 `cnt0` が “0” であれば、つまり変数 `cnt0` が大項目メニューの表示を意味していれば、ステップ S 5 1 1 に進む。これに対して、変数 `cnt0` が “1” であれば、つまり変数 `cnt0` が枝メニューの表示を意味していれば、ステップ S 5 5 5 に進む。

#### 【0143】

ステップ S 5 1 1 では LAN テーブル `<lantable>[0]` に含まれる `LAN_GUI_MUX_NUM` が示す数値 `N` を変数 `max_menu_num` として設定し、続くステップ S 5 1 3 では変数 `k` および `i` を “0” に設定する。ステップ S 5 1 5 では、変数 `i` を変数 `max_menu_num` つまり数値 `N` と比較し、変数 `i` が数値 `N` に達するまでステップ S 5 1 7 ～ S 5 2 9 の処理を繰り返す。

#### 【0144】

ステップ S 5 1 7 では、GUI テーブル `<guitbl>[i]` に含まれる `GUI_VISIBLE` の識別子を判別する。識別子が “表示不可” を示すときはそのままステップ S 5 2 9 に進み、変数 `i` のインクリメントを経てステップ S 5 1 5 に戻る。識別子が “表示可” を示すときは、ステップ S 5 1 9 で GUI テーブル `<guitbl>[i]` に含まれる `GUI_PROPERTY` の識別子を判別する。この識別子が “項目未選択” を示すときはステップ S 5 2 1 に進み、この識別子が “項目選択済み” または “項目ロック” を示すときは、ステップ S 5 2 3 に進む。

#### 【0145】

ステップS 5 2 1では、G U Iテーブル<guitbl>[i]に含まれる\*GUI\_TABLEによって特定される大項目文字列を文字列データGUICONF0.DATから抽出し、抽出した文字列をレジスタdisp\_str[k]に格納する。これによって、大項目文字列が表示可能状態となる。処理が完了すると、ステップS 5 2 7で変数kをインクリメントし、かつステップS 5 2 9で変数iをインクリメントしてからステップS 5 1 5に戻る。

#### 【0 1 4 6】

ステップS 5 2 3では、G U Iテーブル<guitbl>[i]に含まれる\*GUI\_LINKADRのアドレス値を変数menu\_tblとして設定し、G U Iテーブル<guitbl>[i]に含まれるGUI\_MEMBERの第1バイト値を変数tree\_numとして設定し、そして枝メニューテーブル<menu\_tbl>[tree\_num]に含まれる\*str\_tableのアドレス値を変数menu\_strとして設定する。これによって、選択された小項目を管理する枝メニューテーブルの下のメニューストリング<menu\_str>[0]～<menu\_str>[K-1]が注目される。

#### 【0 1 4 7】

ステップS 5 2 5では、注目するメニューストリング<menu\_str>[0]～<menu\_str>[K-1]の中からG U Iテーブル<guitbl>[i]に含まれるGUI\_MEMBERの第2バイト値に対応するメニューストリングを特定し、特定したメニューストリングに含まれるm\_stringによって特定される小項目文字列を文字列データGUICONF0.DATから抽出し、そして抽出した文字列をレジスタdisp\_str[k]に格納する。これによって、選択された小項目の文字列が表示可能状態となる。処理が完了すると、ステップS 5 2 7で変数kをインクリメントし、かつステップS 5 2 9で変数iをインクリメントしてからステップS 5 1 5に戻る。

#### 【0 1 4 8】

ステップS 5 1 5でNOと判断されると、ステップS 5 3 1～S 5 3 5でエラーチェックを行う。ステップS 5 3 1では、変数cnt1が変数max\_menu\_num以上であるかどうか、つまりカーソルC Sの位置に異常があるかどうかを判別する。ステップS 5 3 3では、変数cnt0が大項目メニューの表示を意味する“0”であり、かつG U Iテーブル<guitbl>[cnt1]に含まれるGUI\_SELECTが“選択不可”

の識別子を有するかどうかを判別する。ステップ S 5 3 5 では、<guitbl>[cnt1] に含まれる GUI\_VISIBLE が “表示不可” の識別子を有するかどうかを判別する。

#### 【0149】

ステップ S 5 3 1 ～ S 5 3 5 のいずれか 1 つでも YES と判断されると、エラーが生じているとみなして、ステップ S 5 3 7 で変数 tree\_num を “-1” に設定してから上階層のルーチンに復帰する。これに対して、ステップ S 5 3 1 ～ S 5 3 5 の全てにおいて NO と判断されると、エラーは生じていないとみなして、ステップ S 5 3 9 に進む。

#### 【0150】

ステップ S 5 3 9 では変数 i を “0” に設定し、ステップ S 5 4 1 では変数 i を変数 max\_menu\_num と比較する。変数 i が変数 max\_menu\_num に満たなければ、ステップ S 5 4 3 に進み、レジスタ disp\_str[i] に格納された文字列を画面の i 行目に表示する。表示処理が完了すると、ステップ S 5 4 5 で変数 i をインクリメントしてからステップ S 5 4 1 に戻る。変数 i が変数 max\_menu\_num に達すると、ステップ S 5 4 7 に進み、cnt1 行目の文字列を指向するようにカーソル CS を表示する。

#### 【0151】

ステップ S 5 4 9 では変数 cnt0 が示す数値を判別する。ここで、変数 cnt0 が “0” であれば、つまり変数 cnt0 が大項目メニューの表示を意味していれば、ステップ S 5 5 1 で変数 cnt1 を変数 pre\_cnt1 に設定し、ステップ S 5 5 3 で変数 cnt0 を変数 pre\_cnt0 に設定する。これに対して、変数 cnt0 が “1” であれば、つまり変数 cnt0 が枝メニューの表示を意味していれば、ステップ S 5 5 1 を経ることなく、ステップ S 5 5 3 で変数 cnt0 を変数 pre\_cnt0 に設定する。ステップ S 5 5 3 の処理が完了すると、上階層のルーチンに復帰する。

#### 【0152】

図 3 8 に示すステップ S 5 0 9 で NO と判断されたときは、枝メニューを表示するべく、図 4 2 に示すステップ S 5 5 5 に進む。このステップでは、GUI テーブル <guitbl>[pre\_cnt1] に含まれる \*GUI\_LINKADR のアドレス値を変数 menu\_tbl として設定する。これによって、GUI テーブル <guitbl[pre\_cnt1]> に属す

る枝メニューテーブル<menu\_tbl[0]>~<menu\_tbl[N-1]>が注目される。ステップS557では変数iを“0”に設定し、ステップS559では変数iをGUIテーブル<guitbl[pre\_cnt1]>に含まれるGUI\_MAXNUMが示す数値Iと比較する。そして、変数iが数値Iを下回ればステップS563以降の処理を実行するが、変数iが数値Iに達すれば、エラーが生じたとみなして、ステップS561で変数tree\_\_numを“-1”に設定してから上階層のルーチンに復帰する。

#### 【0153】

ステップS563では、注目する枝メニューテーブル<menu\_tbl[i]>のtree\_\_maxnumが示す数値Lを判別する。数値Lが“0”であれば、注目する枝メニューテーブル<menu\_tbl[i]>の下にはツリーテーブルは全く存在しないとみなし、そのままステップS577に進む。数値Lが“1”以上であれば、注目する枝メニューテーブル<menu\_tbl>[i]の下にツリーテーブル<tree\_tbl>[0]~<tree\_tbl>[L-1]が存在するとみなし、選択された枝メニューに依存するツリーテーブルを検索するべく、ステップS565に進む。

#### 【0154】

ステップS565では、枝メニューテーブル<menu\_tbl>[i]に含まれる\*tree\_\_tableのアドレス値を変数tree\_tblとして設定する。これによって、枝メニューテーブル<menu\_tbl>[i]に属するツリーテーブル<tree\_tbl>[0]~<tree\_tbl>[L-1]が注目される。ステップS567では変数jを“0”に設定し、ステップS569では変数jを枝メニューテーブル<menu\_tbl>[i]のtree\_\_maxnumが示す数値Lと比較する。変数jが数値Lに達したときはステップS571で変数iをインクリメントしてステップS559に戻るが、変数jが数値Lを下回るときは、ステップS573およびS575で上述のステップS413およびS415と同様の処理を行う。

#### 【0155】

ステップS573およびS575のいずれか一方でもNOと判断されると、ツリーテーブルtree\_tbl[j]はGUIテーブル<guitbl>[tree\_tbl[j]. gui\_\_tree[0]]に依存していないとみなし、ステップS577で変数jをインクリメントしてステップS569に戻る。これに対して、ステップS573およびS575の

両方でYESと判断されると、ツリーテーブルtree\_tbl[k]はGUIテーブル<guitbl>[tree\_tbl[k]. gui\_tree[0]]に依存しているとみなし、ステップS579に進む。

#### 【0156】

ステップS579では、選択された枝メニューの番号を有効化すべく、変数iを変数tree\_numに設定する。ステップS581では、枝メニューテーブル<menu\_tbl>[i]に含まれる\*str\_tableのアドレス値を変数menu\_strとして設定し、枝メニューテーブル<menu\_tbl>[i]のstr\_maxnumが示す数値Kを変数max\_menu\_numとして設定する。これによって、枝メニューテーブル<menu\_tbl>[i]に属するメニューストリング<menu\_str>[0]～<menu\_str>[K-1]が注目される。

#### 【0157】

ステップS583では変数iを“0”に設定し、ステップS585では変数iを変数max\_menu\_numつまり数値Kと比較する。変数iが数値K未満のときはステップS587に進み、メニューストリング<menu\_str>[i]の\*menu\_stringによって特定される文字列をレジスタdisp\_str[i]に格納する。この処理を終えると、ステップS589で変数iをインクリメントしてステップS585に戻る。変数iが数値Kに達すると、ステップS531に進む。

#### 【0158】

以上の説明から分かるように、CPU44によって読み取られる表示制御テーブルGUICONF0.TBLは、複数のGUIテーブル<guitbl>[0]～<guitbl>[N-1]とその各々に属する複数のメニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]とを備える。GUIテーブル<guitbl>[0]～<guitbl>[N-1]は、CPU44によって表示処理を施される複数の大項目をそれぞれ管理する。また、メニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]の各々は、CPU44によって表示処理を施される複数の小項目を管理する。

#### 【0159】

メニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]の各々には、自分が属するGUIテーブルと異なるGUIテーブルの下で管理される小項目との依存関係を示すツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[L-1]が割り当てられる。所

望の小項目が選択されると、CPU 44は、このツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[L-1]に基づいて、所望の小項目に依存する複数の小項目を表示する。

#### 【0160】

このように、ツリーテーブル<tree\_tbl>[0]～<tree\_tbl>[L-1]がメニューテーブル<menu\_tbl>[0]～<menu\_tbl>[I-1]の各々に割り当てられるため、CPU 44は、小項目を的確に画面に表示することができる。つまり、デジタルカメラ10は異なるメニュー画像を共通の手順で画面に表示することができる。

#### 【0161】

また、所望の小項目を表す枝メニュー番号および小項目番号つまり所望小項目番号が、所望の小項目を管理するGUIテーブルにGUI\_MEMBERとして記述される。CPU 44は、かかる所望小項目情報に基づいて、所望の小項目に対応する大項目の代わりに所望の小項目を表示する。これによって、いずれの小項目が選択されたかを容易に把握することができ、操作性が向上する。なお、メニューテーブルには複数のツリーテーブルを割り当てることができ、これによってメニューテーブルの総数ひいてはデータサイズの抑制が可能となる。

#### 【0162】

さらに、所望の小項目の選択が解除されると、GUI\_MEMBERに記述された枝メニュー番号および小項目番号が、項目未選択を示す“－1”に切り換えられる。プロセサは、かかる小項目未選択情報に基づいて、所望の小項目の代わりに所望の小項目に対応する大項目を表示する。

#### 【0163】

さらにまた、所望の小項目が選択されていないメニューテーブルの小項目に依存するメニューテーブルが属するGUIテーブルのGUI\_SELECTに、“選択不可”を示す識別子つまり選択不可情報が記述される。CPU 44は、かかる選択不可情報が割り当てられたGUIテーブルが管理する大項目の表示を中止する。これによって、未選択の小項目に関連する大項目の表示が中止され、操作性が向上する。

#### 【0164】

また、GUI テーブル  $\langle \text{guitbl} \rangle[0] \sim \langle \text{guitbl} \rangle[N-1]$  の各々に属する複数のメニューテーブル  $\langle \text{menu\_tbl} \rangle[0] \sim \langle \text{menu\_tbl} \rangle[I-1]$  は連続し、GUI テーブル  $\langle \text{guitbl} \rangle[0] \sim \langle \text{guitbl} \rangle[N-1]$  の各々には \*GUI\_LINKADR および GUI\_MAXNUM が記述される。これによって、メニューテーブル毎にアドレス情報を記述する必要がなくなり、データサイズの抑制が可能となる。

#### 【0165】

なお、この実施例では、メモ리카ード 38 および通信カード 40 がスロット 36 に選択的に装着されるが、デュアルスロットを用意すれば、メモ리카ード 38 および通信カード 40 を同時に装着できる。また、通信機能付きメモ리카ードを用意すれば、単一のスロットでメモリ機能および通信機能を同時に実現できる。

#### 【0166】

また、この実施例では、表示制御テーブル GUICONF0.TBL および文字列データ GUICONF0.DAT、ならびに通信不能時の画像ファイルを内蔵型のフラッシュメモリ 42 に格納するようにしている。しかし、上述のデュアルスロットを用意すれば、これらのデータはメモ리카ード 38 に格納してもよく、通信機能付きメモ리카ードを用意すれば、これらのデータは通信機能付きメモ리카ードに格納してもよい。

#### 【0167】

さらに、この実施例では、楽器の製作現場の詳細情報を入力するのに適したメニュー画像を想定しているが、これに代えてビルの建設現場の詳細情報を入力するためのメニュー画像を用意すれば、建設会社向けのデジタルカメラが得られる。また、交通事故現場の詳細情報を入力するためのメニュー画像を用意すれば、損害保険会社向けのデジタルカメラが得られる。

#### 【0168】

また、この実施例では、デジタルカメラを用いて説明したが、この発明は、メニュー画像を表示するあらゆる電子機器に適用できることは言うまでもない。

#### 【図面の簡単な説明】

#### 【図 1】

この発明の一実施例を示すブロック図である。

**【図 2】**

図 1 実施例に適用されるメニュー構造の一部を示す図解図である。

**【図 3】**

図 1 実施例に適用されるメニュー構造の他の一部を示す図解図である。

**【図 4】**

図 1 実施例に適用されるメニュー構造のその他の一部を示す図解図である。

**【図 5】**

メニュー構造を形成する要素の名称を記述した図解図である。

**【図 6】**

画面表示の遷移の一例を示す図解図である。

**【図 7】**

(A) は画面に表示されるメニューの一例を示す図解図であり、(B) は画面に表示されるメニューの他の一例を示す図解図である。

**【図 8】**

(A) は画面に表示されるメニューのその他の一例を示す図解図であり、(B) は画面に表示されるメニューのさらにその他の一例を示す図解図である。

**【図 9】**

(A) は画面に表示されるメニューの他の一例を示す図解図であり、(B) は画面に表示されるメニューのその他の一例を示す図解図である。

**【図 10】**

(A) は画面に表示されるメニューのさらにその他の一例を示す図解図であり、(B) は画面に表示されるメニューの他の一例を示す図解図である。

**【図 11】**

(A) は画面に表示されるメニューのその他の一例を示す図解図であり、(B) は画面に表示されるメニューのさらにその他の一例を示す図解図である。

**【図 12】**

(A) は画面に表示されるメニューの他の一例を示す図解図であり、(B) は画面に表示されるメニューのその他の一例を示す図解図である。

**【図 13】**



(A) は画面に表示されるメニューのさらにその他の一例を示す図解図であり、  
(B) は画面に表示されるメニューの他の一例を示す図解図である。

【図 1 4】

(A) は画面に表示されるメニューのその他の一例を示す図解図であり、(B) は画面に表示されるメニューのさらにその他の一例を示す図解図である。

【図 1 5】

図 2 ～図 4 に示すメニュー構造に対応する表示制御テーブルの一例を示す図解図である。

【図 1 6】

図 1 5 に示す表示制御テーブルの構造の一例を示す図解図である。

【図 1 7】

図 2 ～図 4 に示すメニュー構造に対応する文字列データの一例を示す図解図である。

【図 1 8】

図 1 実施例に適用される S D R A M のマッピング状態の一例を示す図解図である。

【図 1 9】

図 1 実施例に適用されるフラッシュメモリのマッピング状態の一例を示す図解図である。

【図 2 0】

図 1 実施例の動作の一部を示すフロー図である。

【図 2 1】

図 1 実施例の動作の他の一部を示すフロー図である。

【図 2 2】

図 1 実施例の動作のその他の一部を示すフロー図である。

【図 2 3】

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

【図 2 4】

図 1 実施例の動作の他の一部を示すフロー図である。

**【図 2 5】**

図 1 実施例の動作のその他の一部を示すフロー図である。

**【図 2 6】**

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

**【図 2 7】**

図 1 実施例の動作の他の一部を示すフロー図である。

**【図 2 8】**

図 1 実施例の動作のその他の一部を示すフロー図である。

**【図 2 9】**

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

**【図 3 0】**

図 1 実施例の動作の他の一部を示すフロー図である。

**【図 3 1】**

図 1 実施例の動作のその他の一部を示すフロー図である。

**【図 3 2】**

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

**【図 3 3】**

図 1 実施例の動作の他の一部を示すフロー図である。

**【図 3 4】**

図 1 実施例の動作のその他の一部を示すフロー図である。

**【図 3 5】**

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

**【図 3 6】**

図 1 実施例の動作の他の一部を示すフロー図である。

**【図 3 7】**

図 1 実施例の動作のその他の一部を示すフロー図である。

**【図 3 8】**

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

**【図 3 9】**

図 1 実施例の動作の他の一部を示すフロー図である。

【図 40】

図 1 実施例の動作のその他の一部を示すフロー図である。

【図 41】

図 1 実施例の動作のさらにその他の一部を示すフロー図である。

【図 42】

図 1 実施例の動作の他の一部を示すフロー図である。

【図 43】

図 1 実施例の動作のその他の一部を示すフロー図である。

【符号の説明】

10…デジタルカメラ

12…イメージセンサ

20…メモリコントローラ

22…SDRAM

24…JPEGコーデック

30…LCD

32…キャラクタジェネレータ

40…通信カード

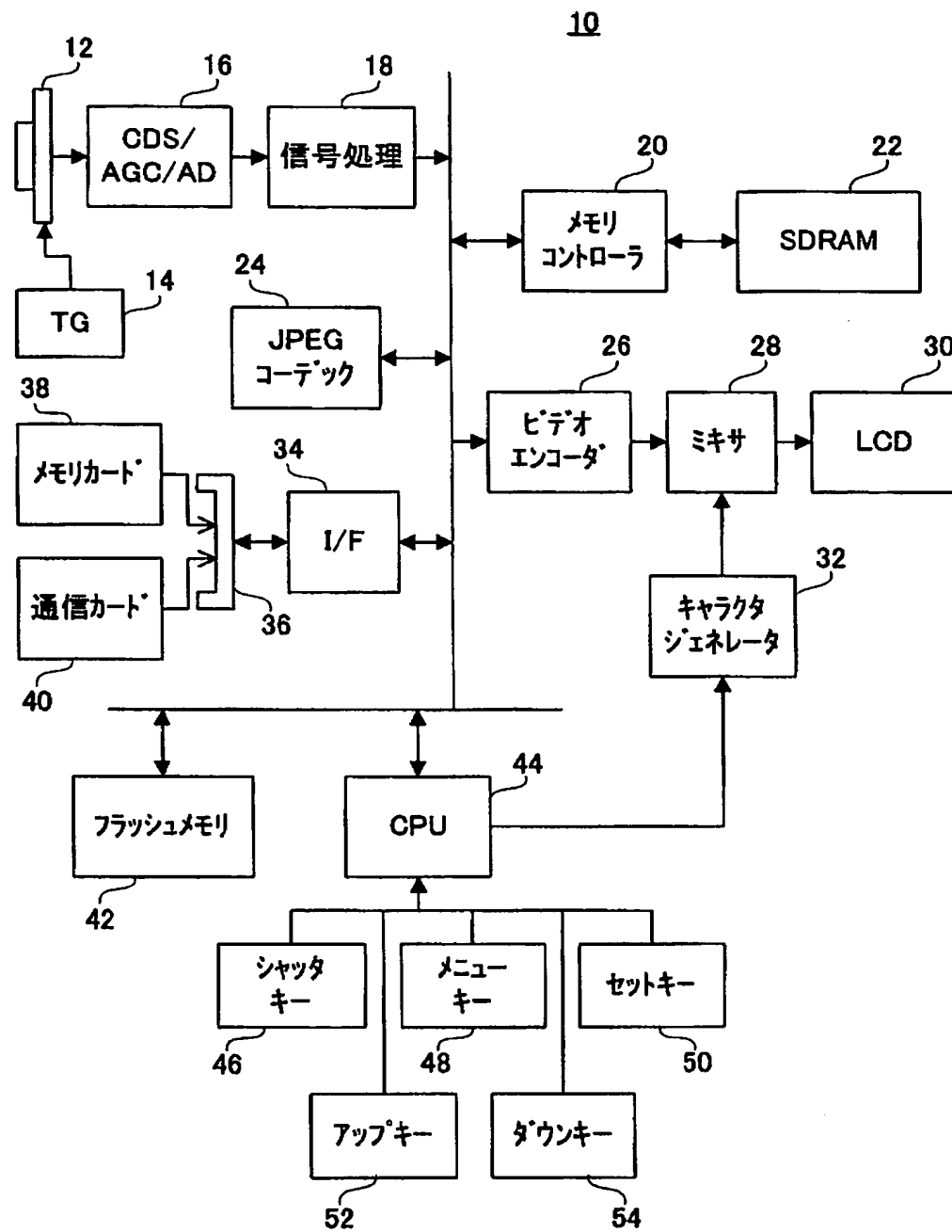
42…フラッシュメモリ

44…CPU

【書類名】

図面

【図1】



【図 2】

0		音楽ジャンル	依存関係			1	楽器分類	依存関係			2	楽器名	依存関係							
			0					0					0							
0	0	クラシック	-1	-1	-1	0	0	弦楽器	-1	-1	-1	0	0	バイオリン	1	0	0			
							1	木管楽器										1	0	0
	1	ジャズ																		
	2	その他																		
							0	2	金管楽器				1	0	オーボエ	1	0	1		
							3	打楽器				1		フルート						
							4	鍵盤楽器				2		クラリネット						
									3	ファゴット										
												2	0	サクソフォン						
													1	ホルン						
													2	トランペット						
													3	トロンボーン						
												4	ユーフォニアム							
													4	チューバ						
												3	0	ティンパニ	1	0	3			
													1	スネアドラム						
													2	バースドラム						
												4	0	ピアノ	1	0	4			
													1	チェレスタ						
													2	オルガン						

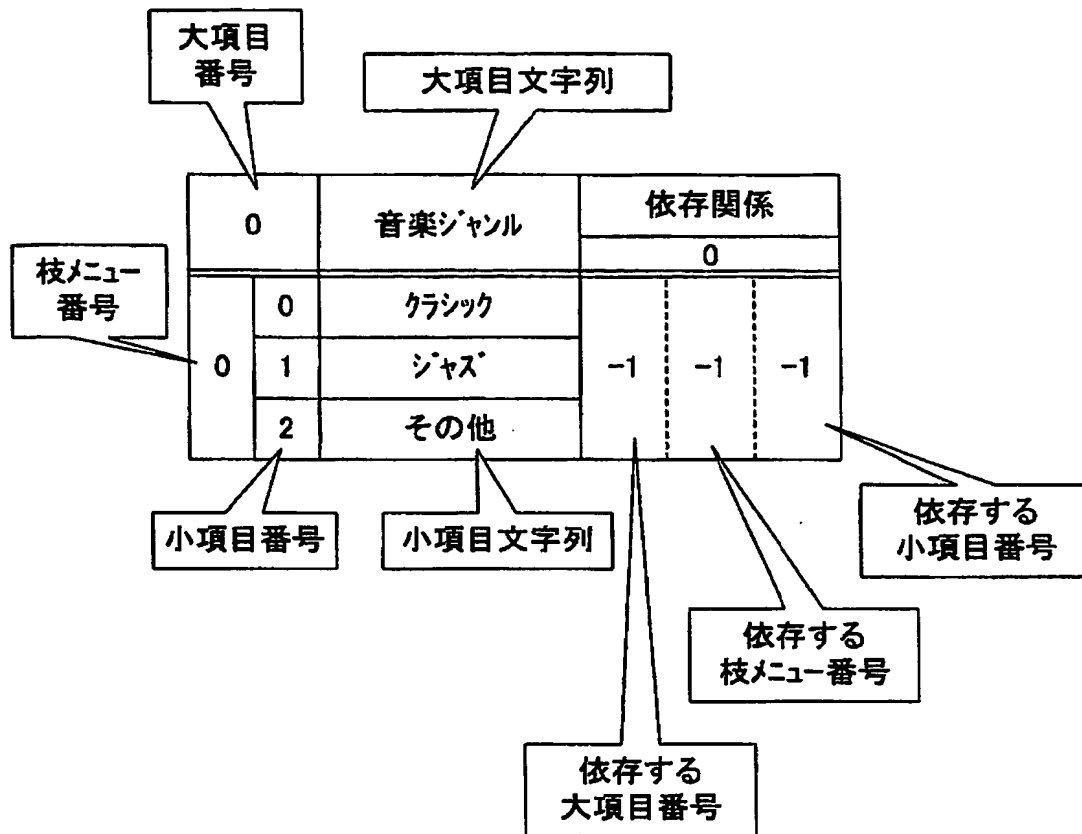
【図3】

3	材質	依存関係	依存関係	依存関係	依存関係	依存関係	依存関係	依存関係
		0	1	2	3	4	5	6
0	0 木材	2 0 0	2 0 1	2 0 2	2 0 3	2 1 0	2 1 2	2 1 3
	1 その他							
1	0 金属							
	1 木材	2 1 1						
	2 その他							
2	0 金属	2 1 4	2 2 0	2 2 1	2 2 2	2 2 3	2 2 4	
	1 その他							

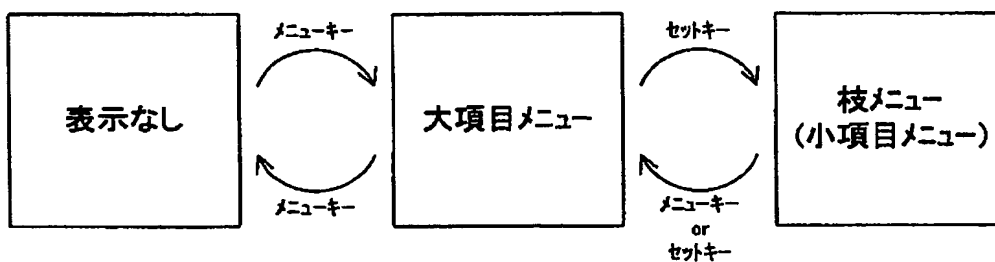
【図4】

4	材料	依存関係	依存関係	5	表面処理	依存関係	依存関係
		0	1			0	1
0	0 真鍮(銅70%)	3 1 0	3 2 0	0	0 無垢	3 1 0	3 2 0
	1 真鍮(銅80%)				1 ラッカー		
	2 真鍮(銅90%)				2 その他塗装		
	3 金				3 金メッキ		
	4 銀				4 銀メッキ		
	5 その他				5 その他メッキ		
1	0 マホガニ(紅木)	3 0 0	3 1 1	1	0 無垢	3 0 0	3 1 1
	1 エホニ(黒檀)				1 ラッカー		
	2 ロースウツ(紫檀)				2 漆		
	3 その他				3 ニス		
					4 その他塗装		

【図 5】

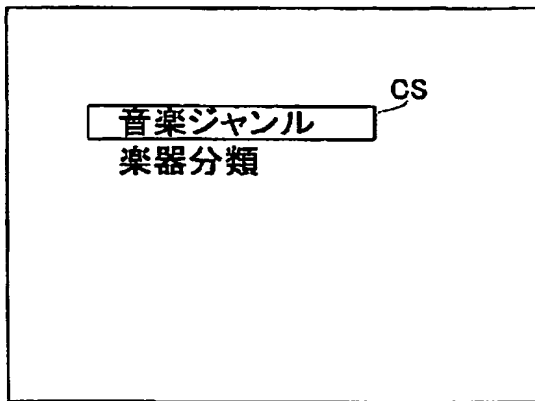


【図 6】

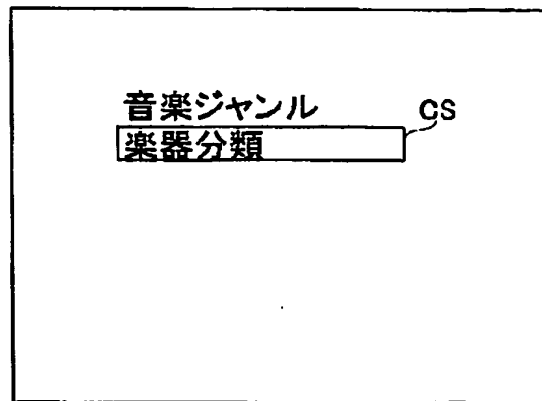


【図 7】

(A)

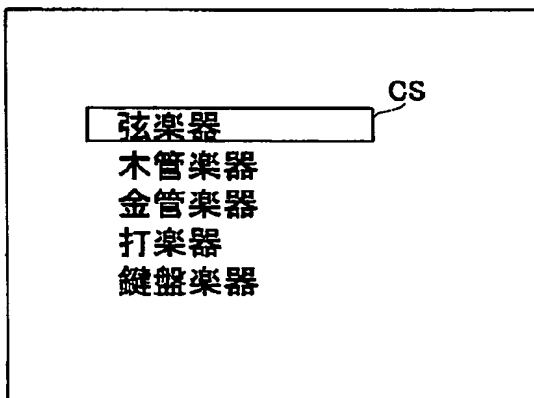


(B)

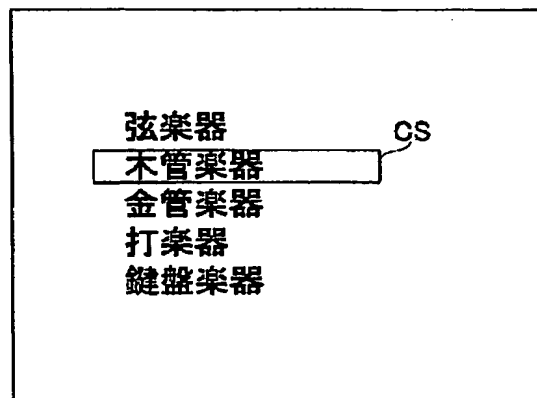


【図 8】

(A)

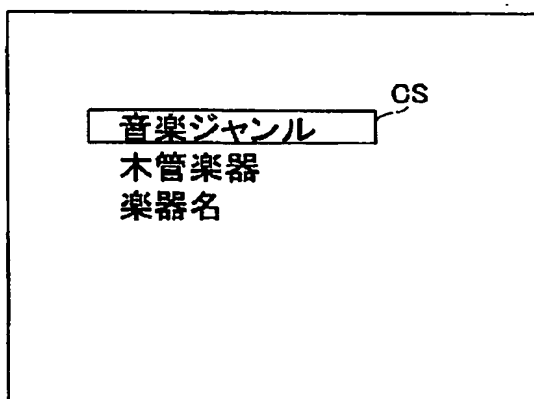


(B)

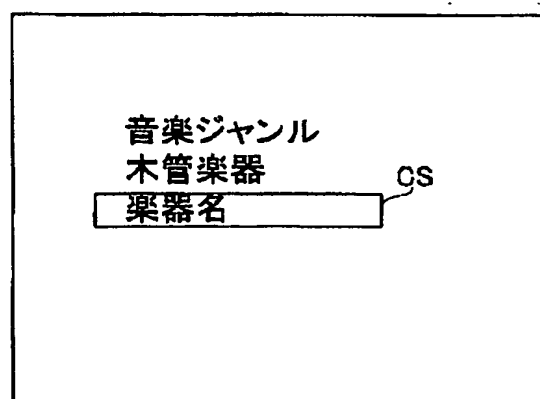


【図 9】

(A)



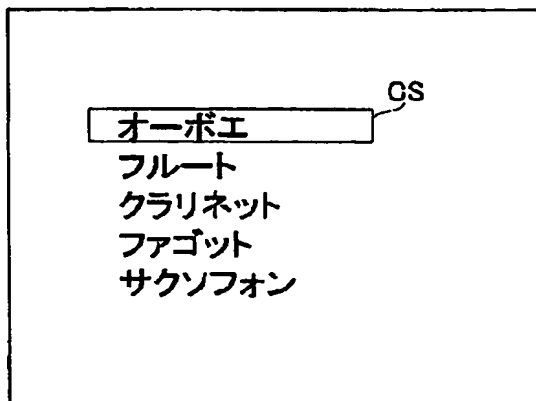
(B)



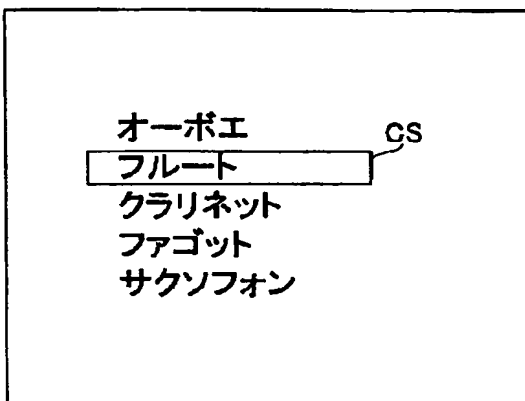


【図 10】

(A)

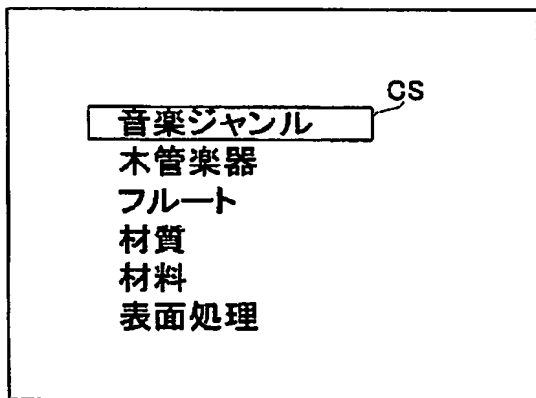


(B)

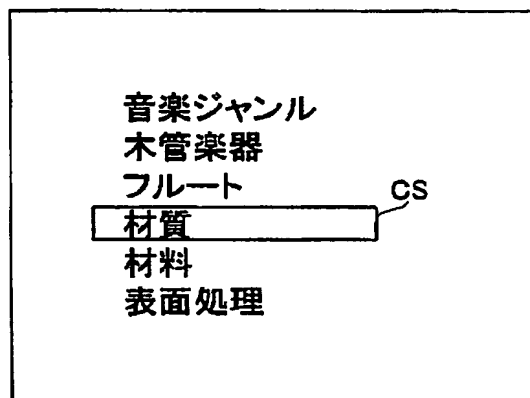


【図 11】

(A)

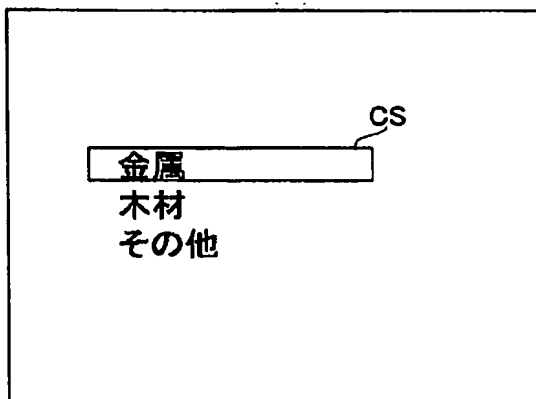


(B)

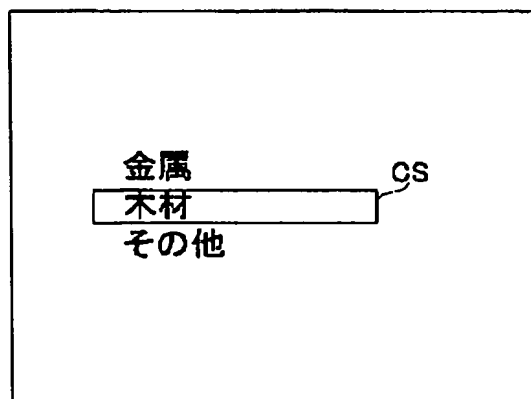


【図 12】

(A)

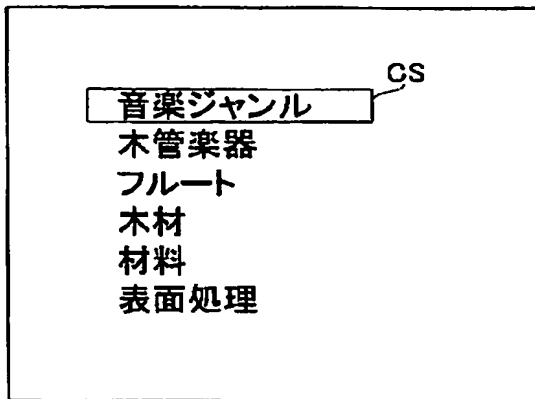


(B)

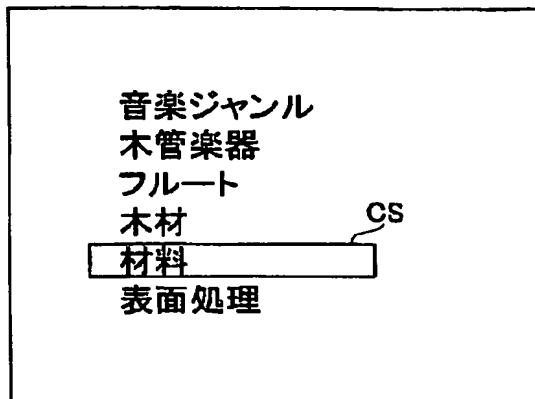


【図 13】

(A)

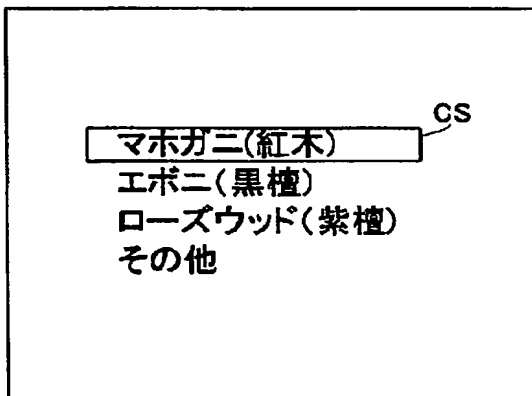


(B)

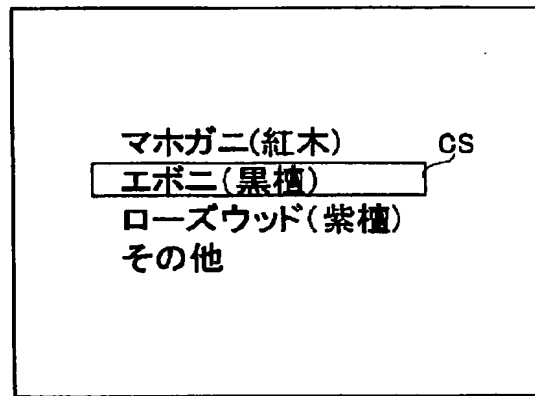


【図 14】

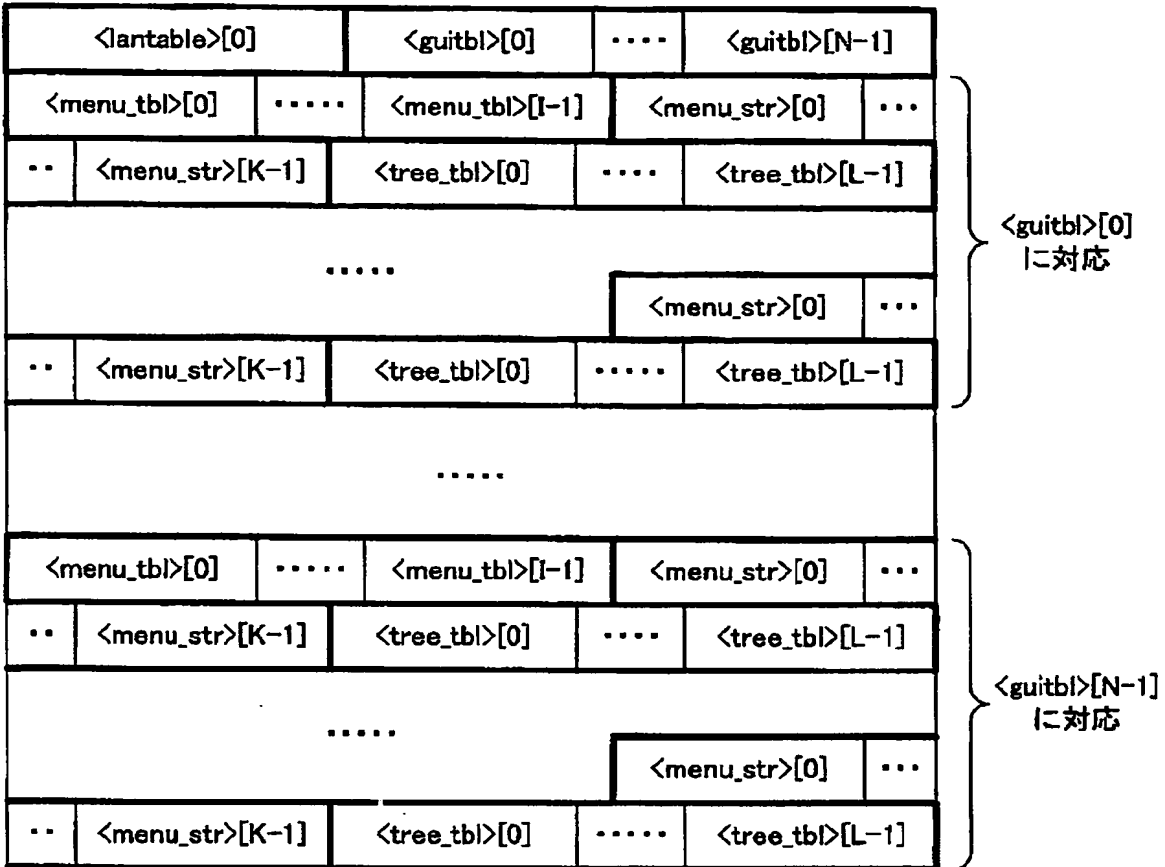
(A)



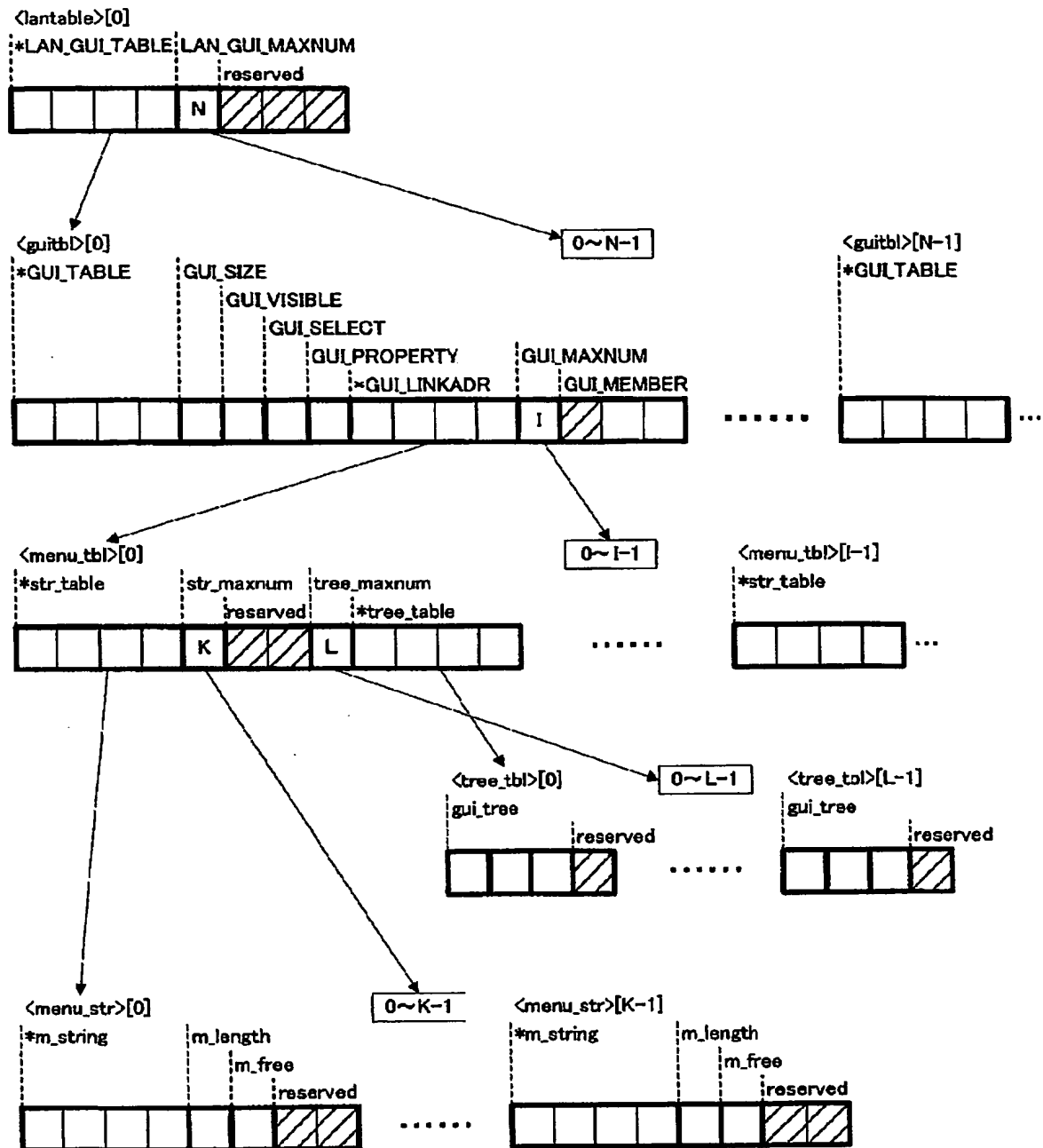
(B)



【図 15】



【図 16】

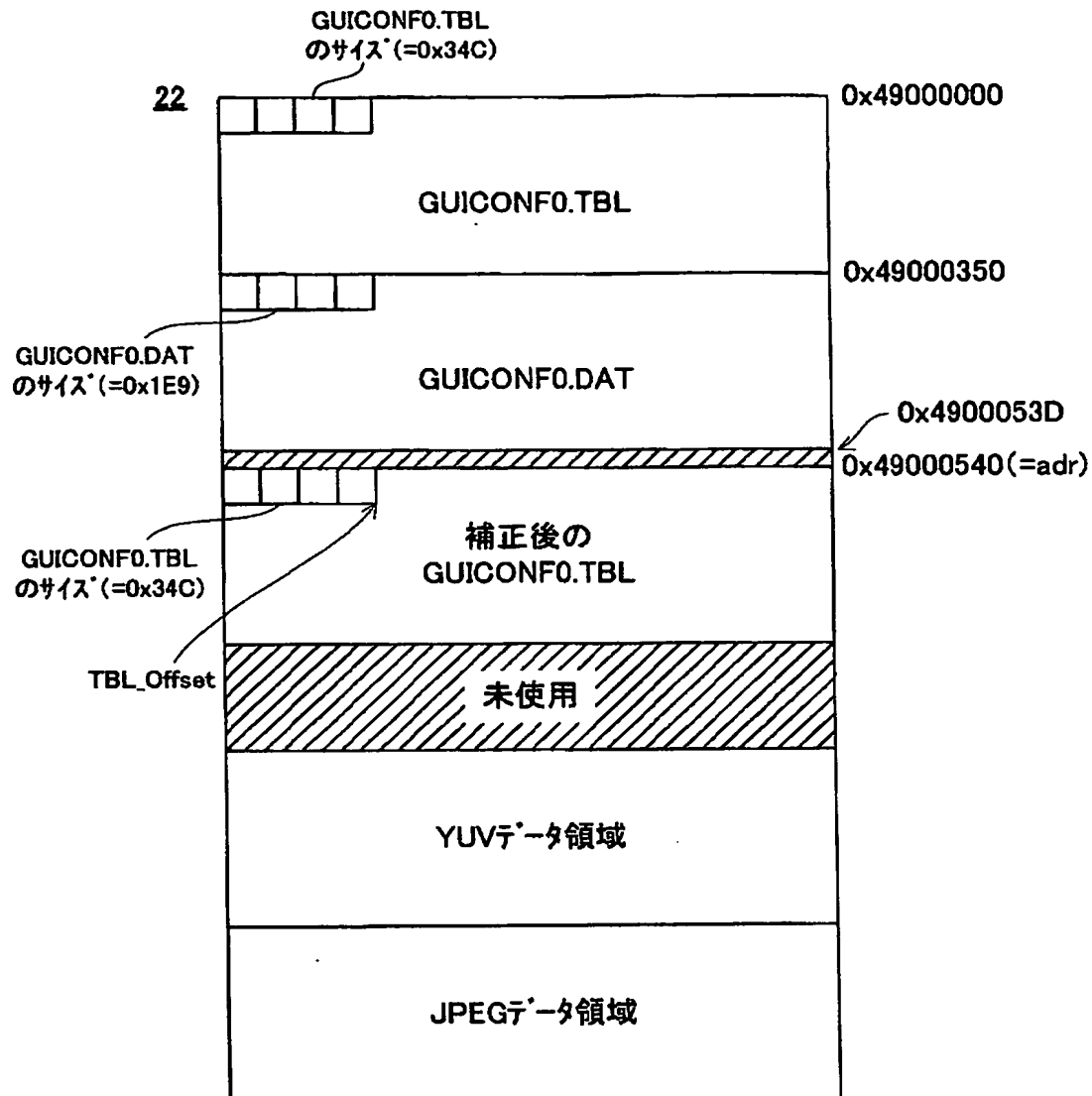


【図17】

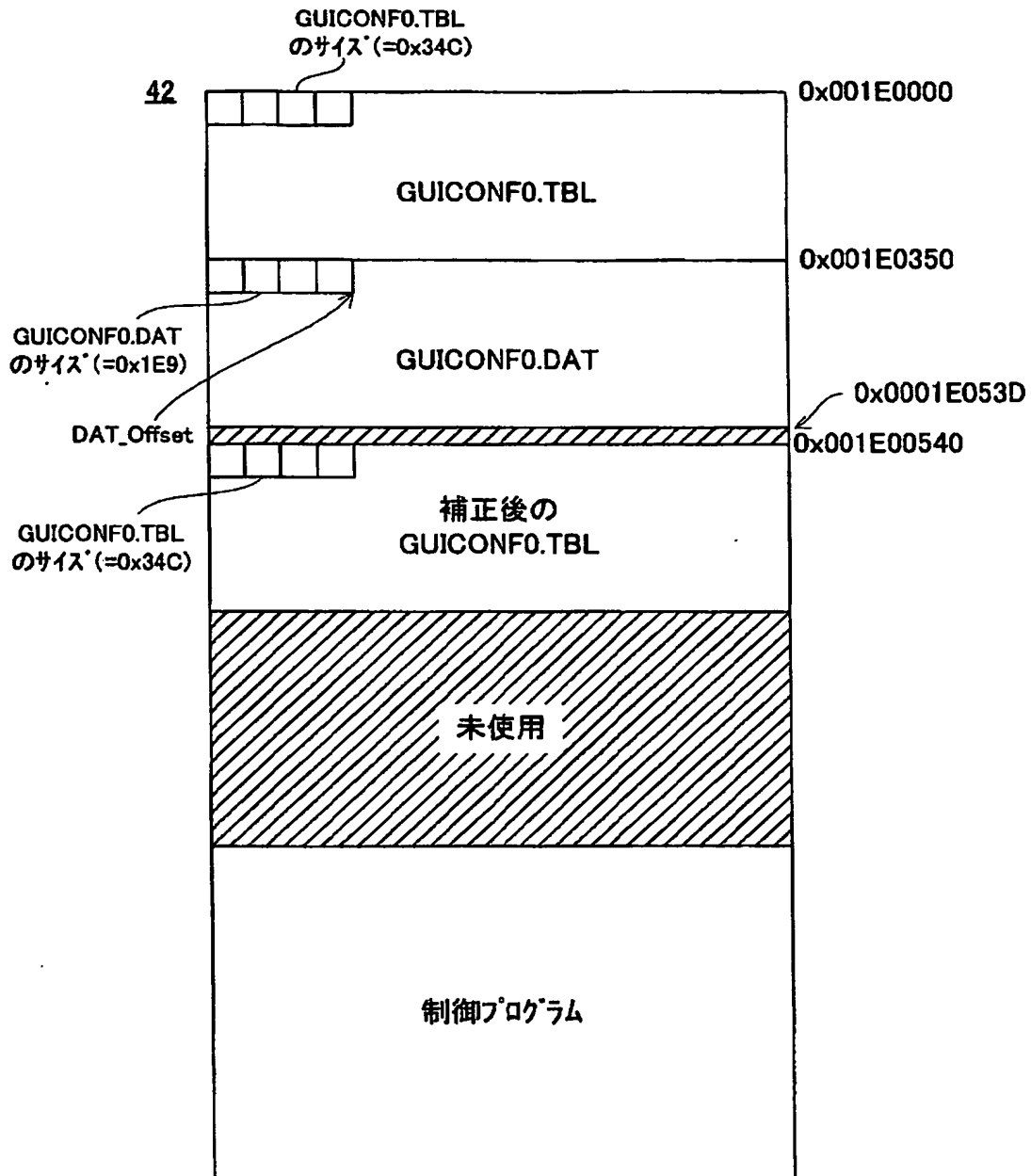
音	楽	ジ	ャ	ン	ル	00	楽	器	分	類	00	楽	器	名					
00	材	質	00	材	料	00	表	面	処	理	00	ク	ラ	シ	ッ				
ク	00	ジ	ャ	ズ	00	そ	の	他	00	弦	楽	器	00	木	管				
楽	器	00	金	管	楽	器	00	打	楽	器	00	鍵	盤	楽	器				
00	バ	イ	オ	リ	ン	00	ビ	オ	ラ	00	チ	エ	ロ	00	コ				
	ン	ト	ラ	バ	ス	00	オ	ー	ポ	エ	00	フ	ル	ー	ト				
00	ク	ラ	リ	ネ	ッ	ト	00	フ	ァ	ゴ	ッ	ト	00	サ					
ク	ソ	フ	ォ	ン	00	ホ	ル	ン	00	ト	ラ	ン	ペ	ッ					
ト	00	ト	ロ	ン	ボ	ー	ン	00	ユ	ー	フ	ォ	ニ	ア					
ム	00	チ	ユ	ー	バ	00	テ	ィ	ン	パ	ニ	00	ス	ネ	ア				
	ド	ラ	ム	00	バ	ス	ド	ラ	ム	00	ピ	ア	ノ	00	チ				
エ	レ	ス	タ	00	オ	ル	ガ	ン	00	木	材	00	金	属	00				
真	鍍	(	銅	7	0	%	)	00	真	鍍	(	銅	8	0	%	)	00	真	鍍
(	銅	9	0	%	)	00	金	00	銀	00	マ	ホ	ガ	ニ	(	紅	木	)	
00	エ	ポ	ニ	(	黒	檀	)	00	ロ	ー	ズ	ウ	ッ	ド	(	紫			
	檀	)	00	無	垢	00	ラ	ッ	カ	ー	00	そ	の	他	塗	装			
00	金	メ	ッ	キ	00	銀	メ	ッ	キ	00	そ	の	他	メ					
ッ	キ	00	漆	00	ニ	ス	00												

00:文字の区切りを表すnullコード

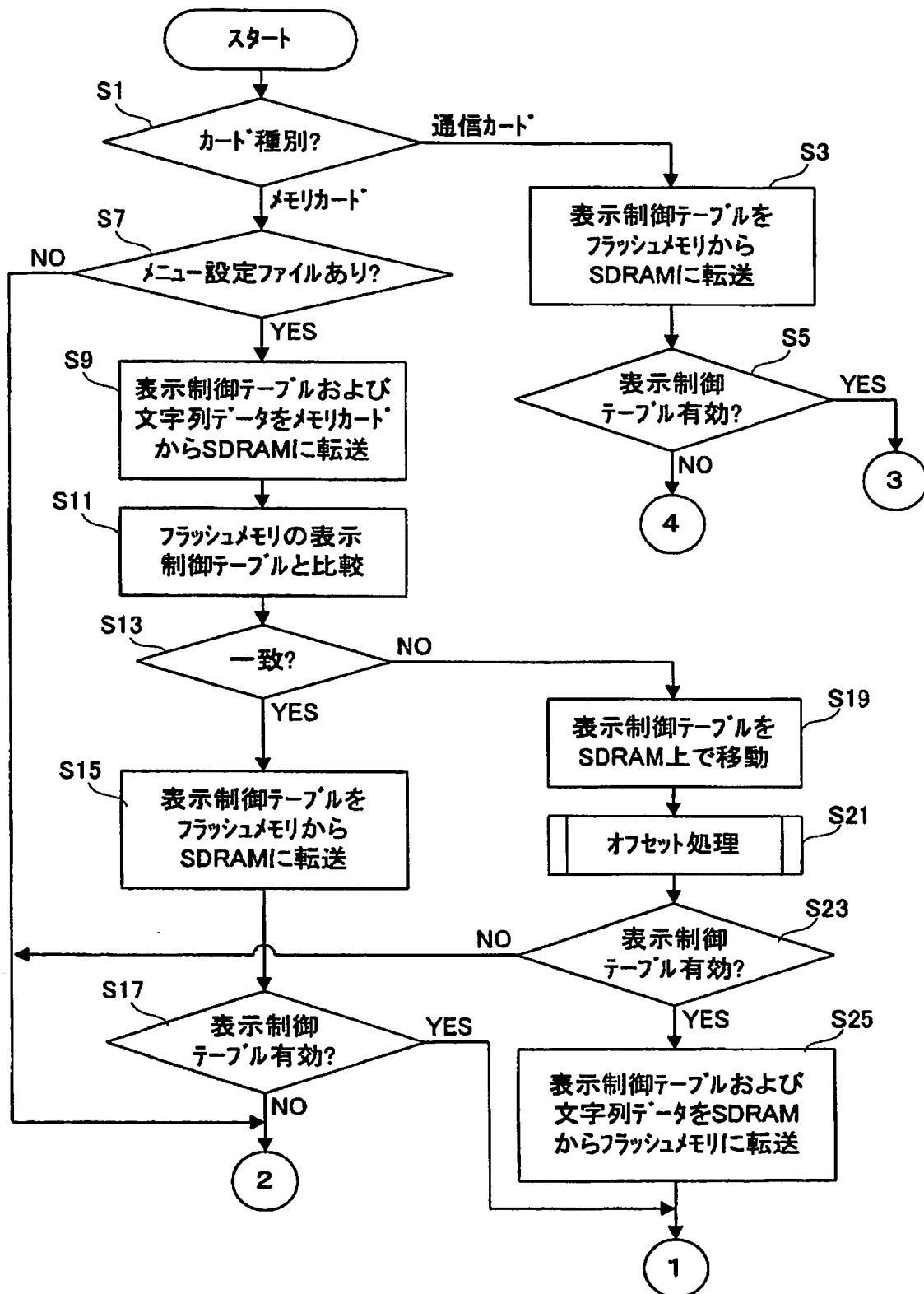
【図 18】



【図 19】

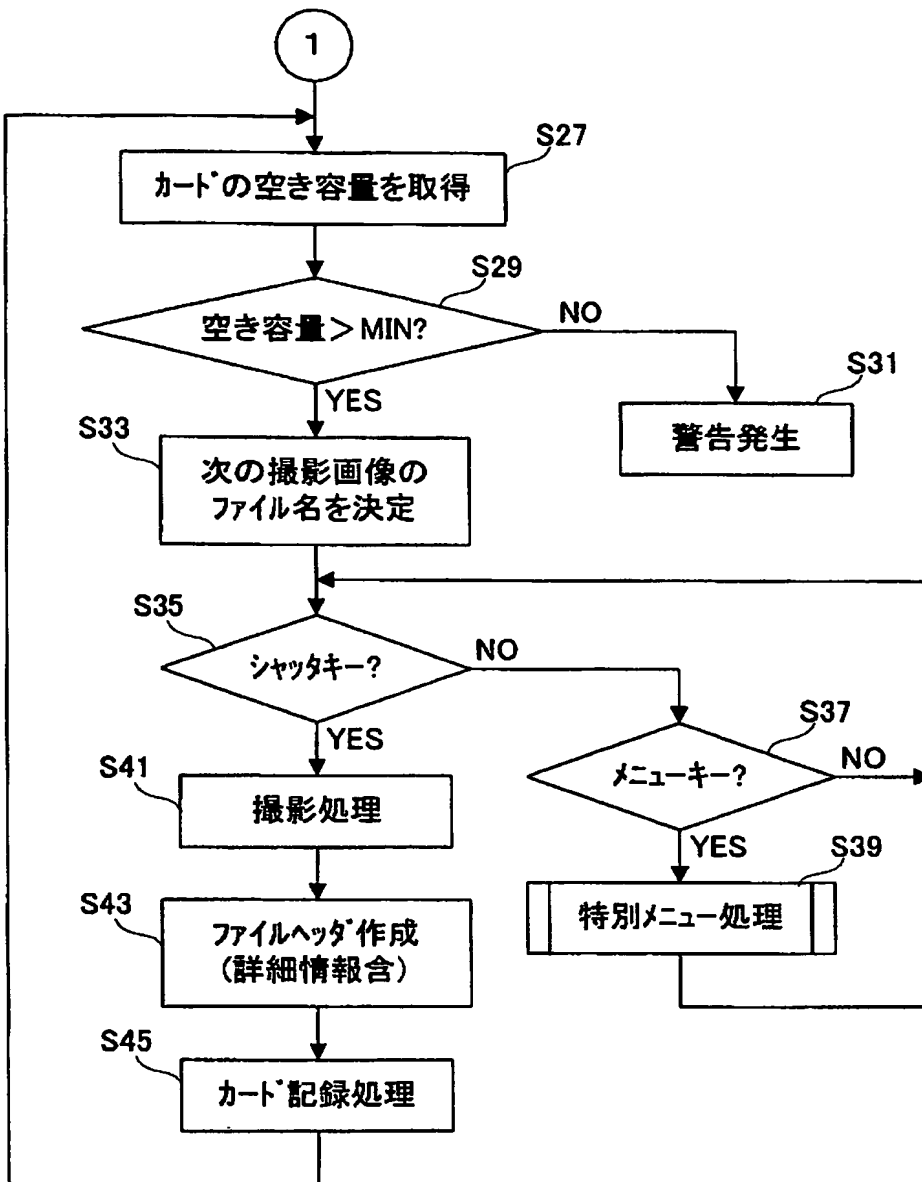


【図 20】

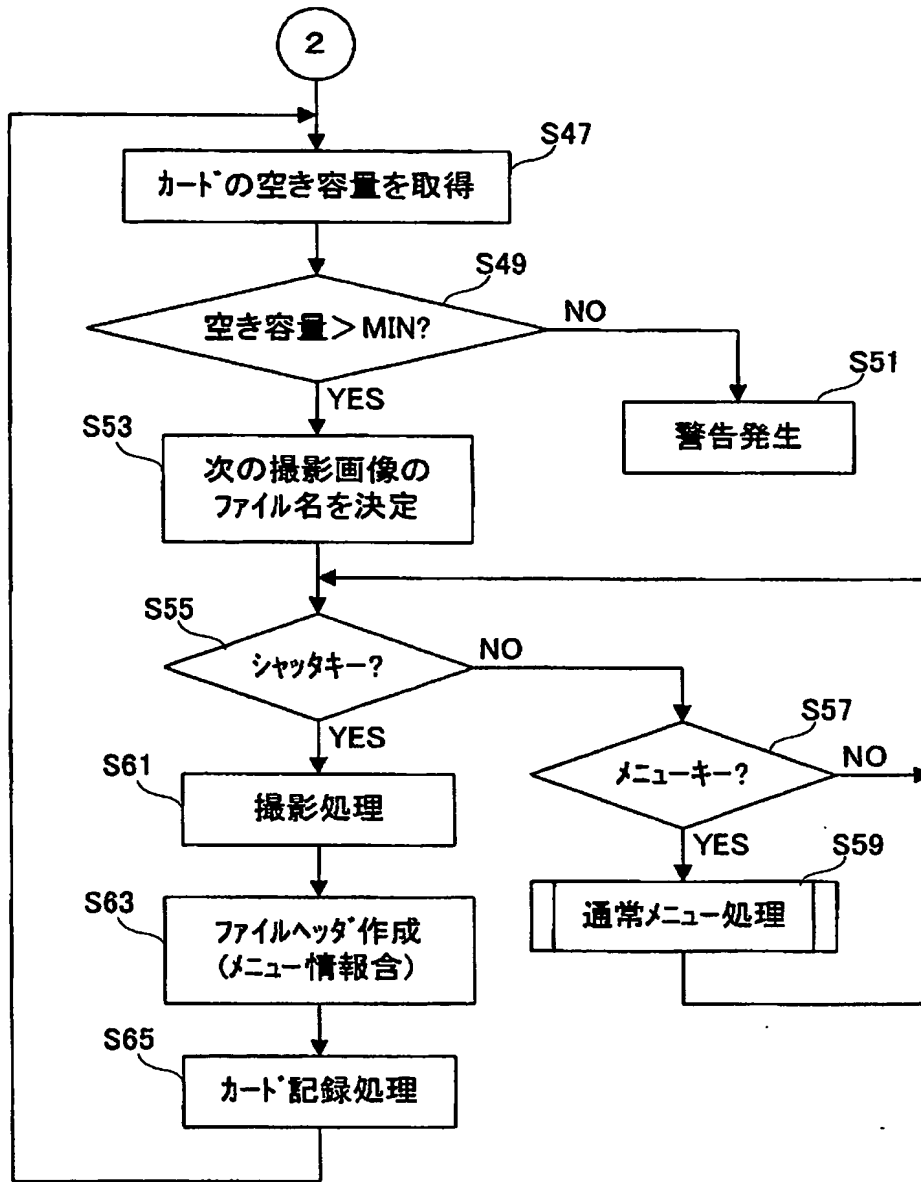




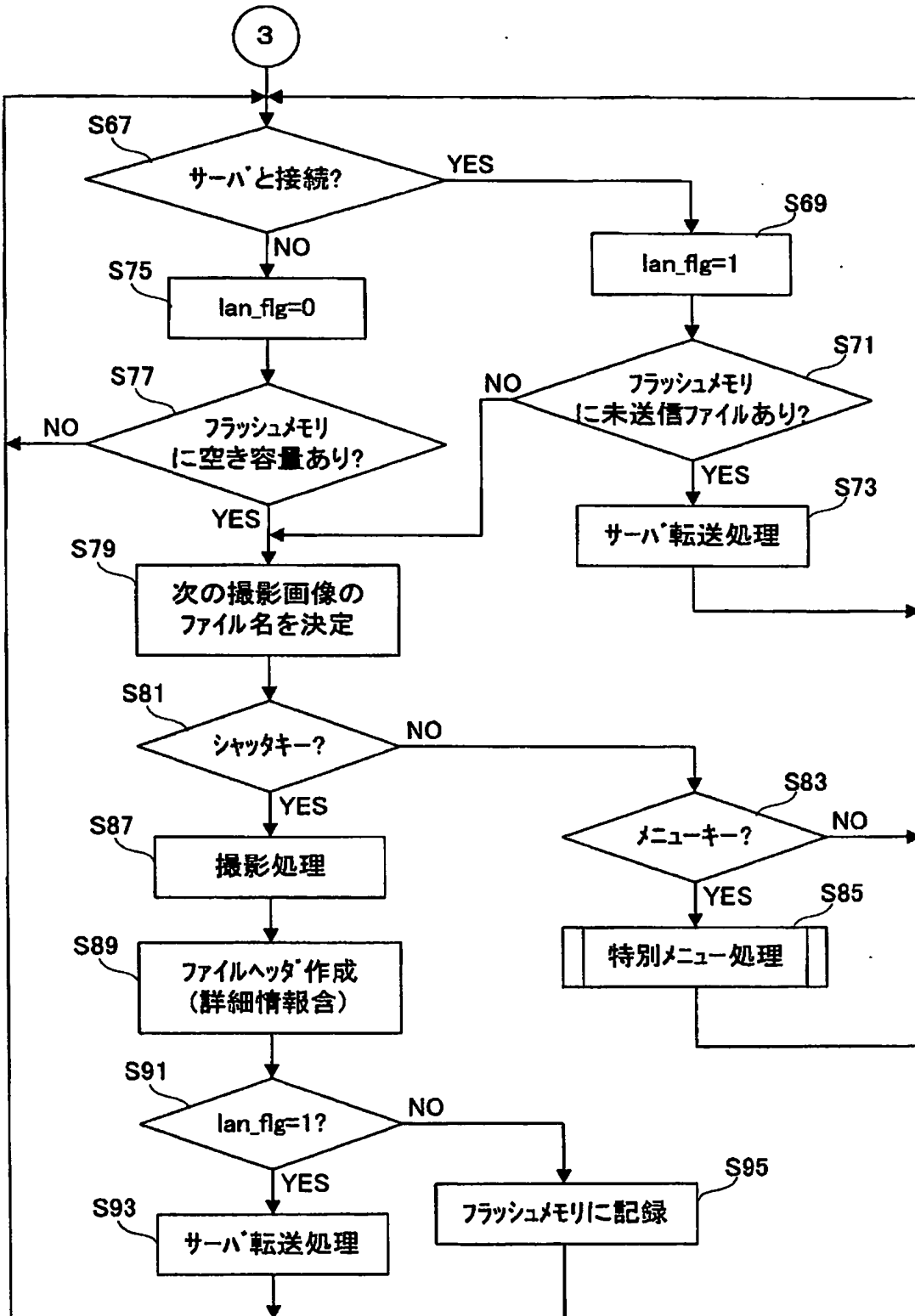
【図 21】



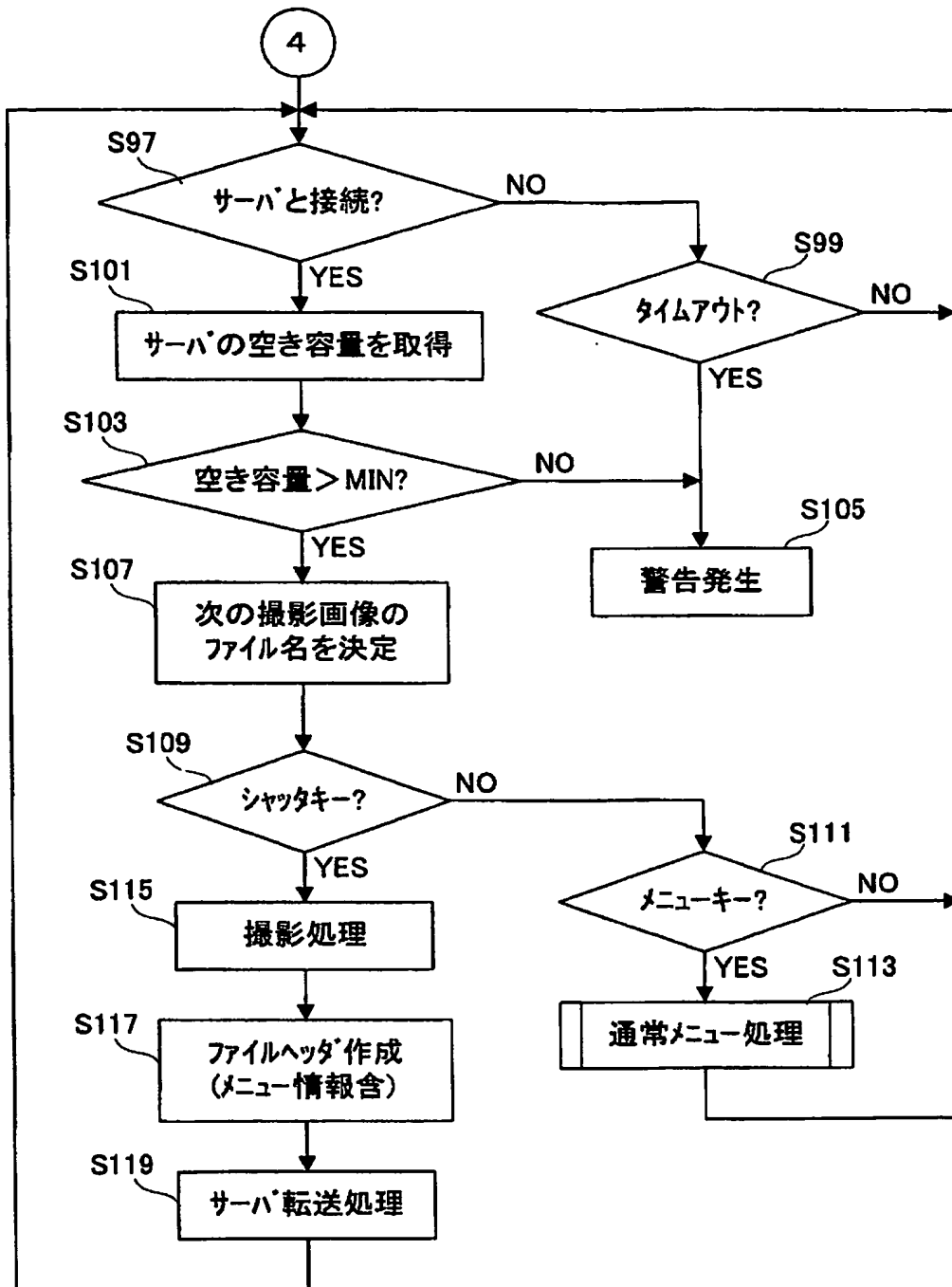
【図 22】



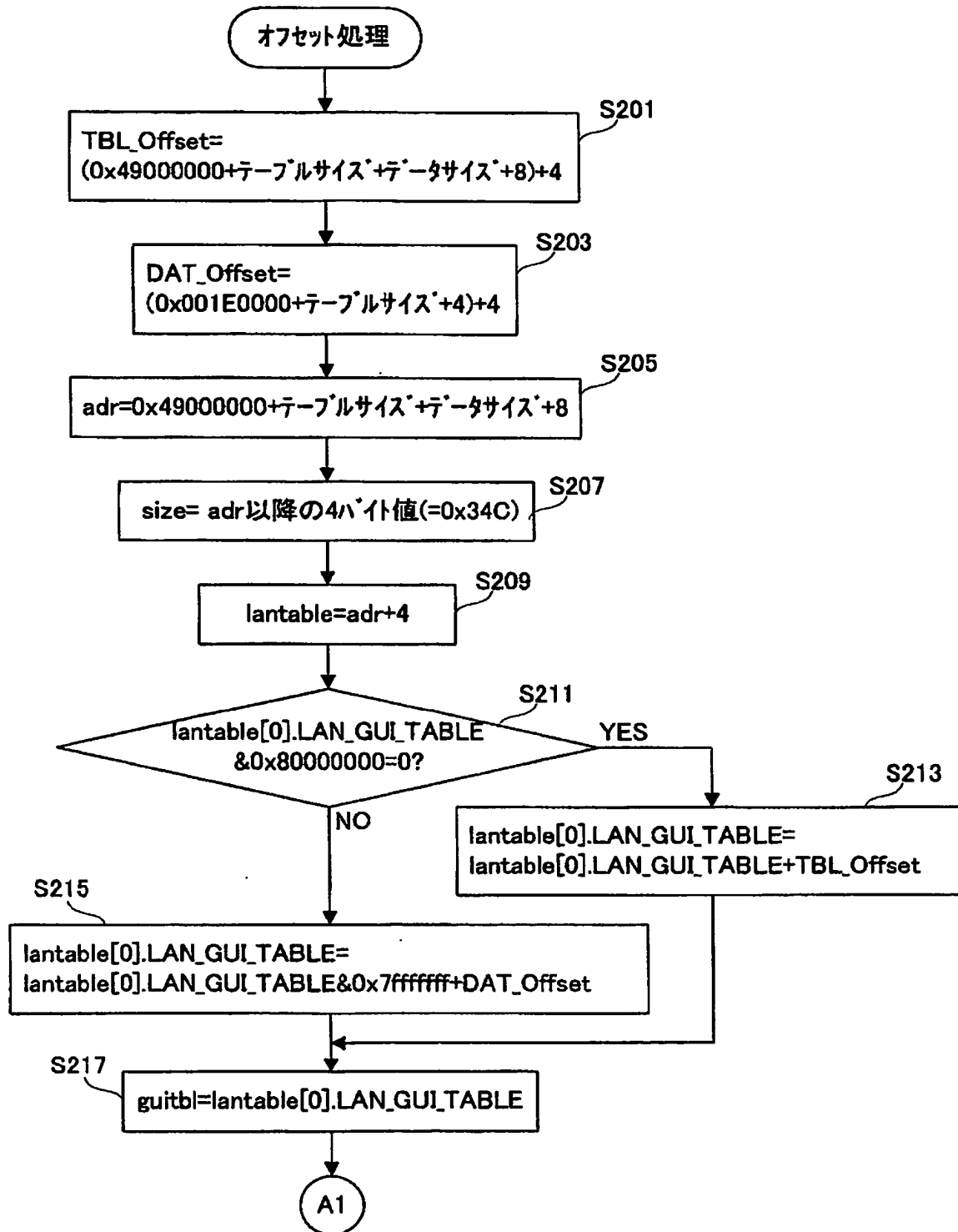
【図23】



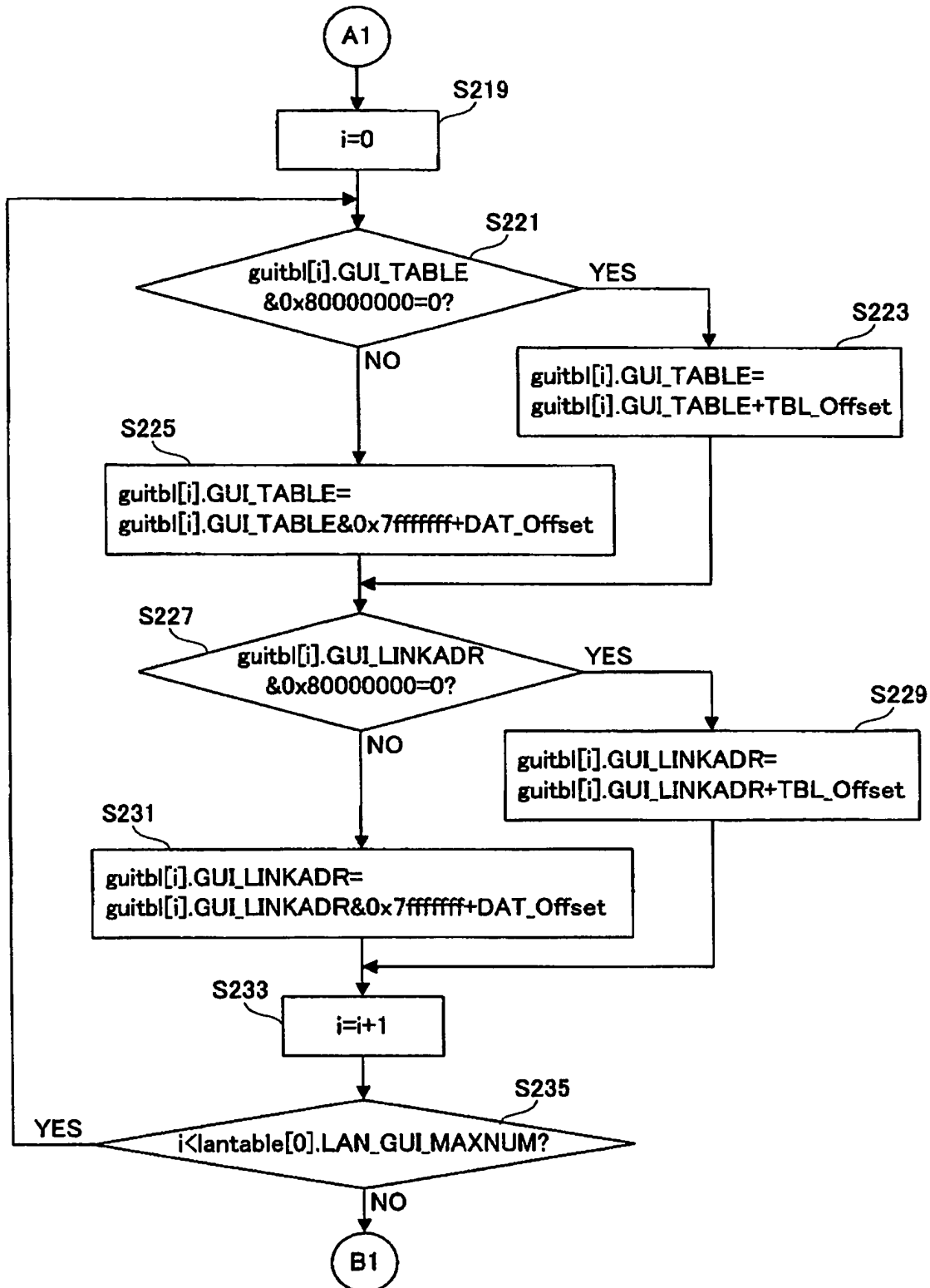
【図 24】



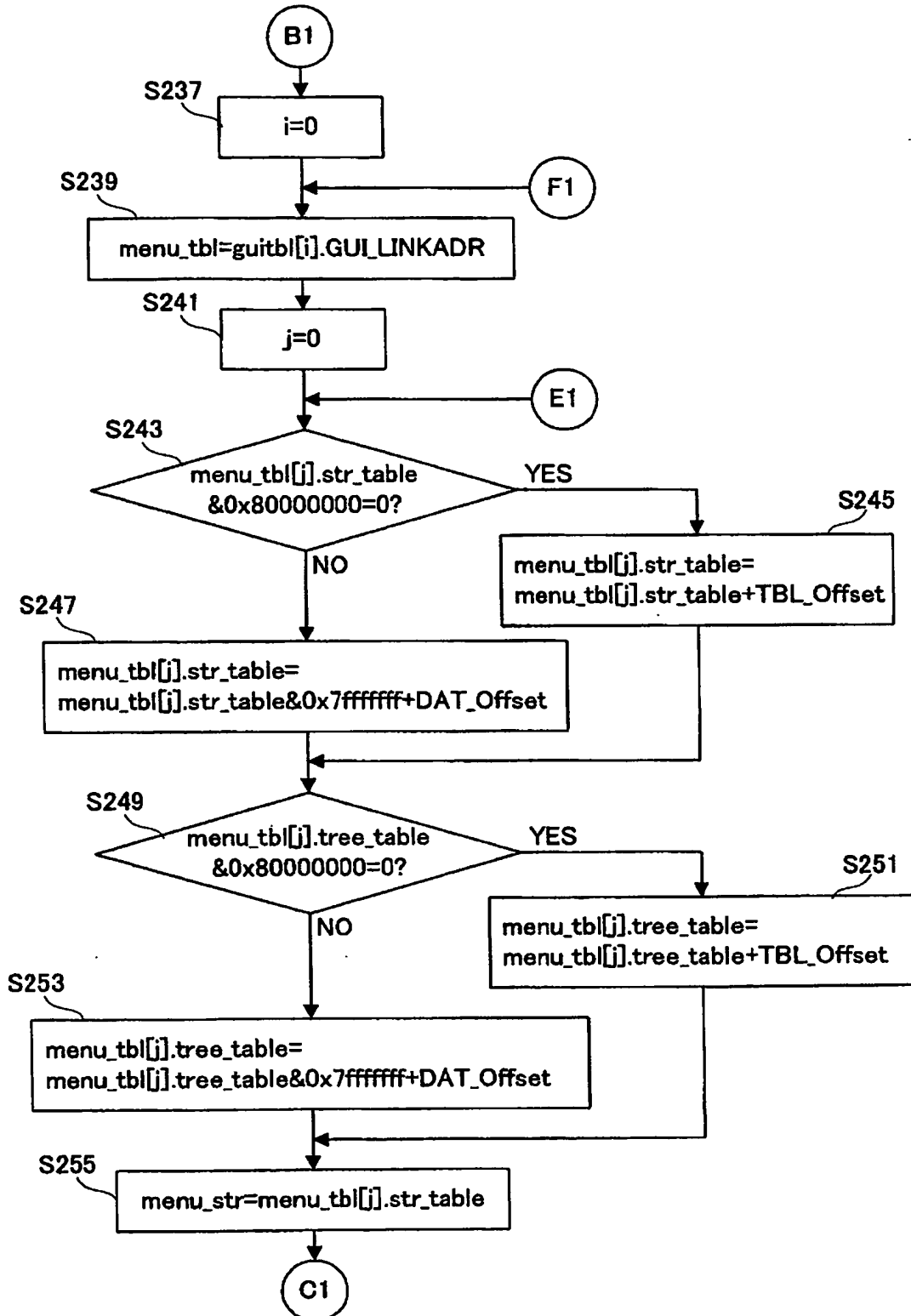
【図 25】



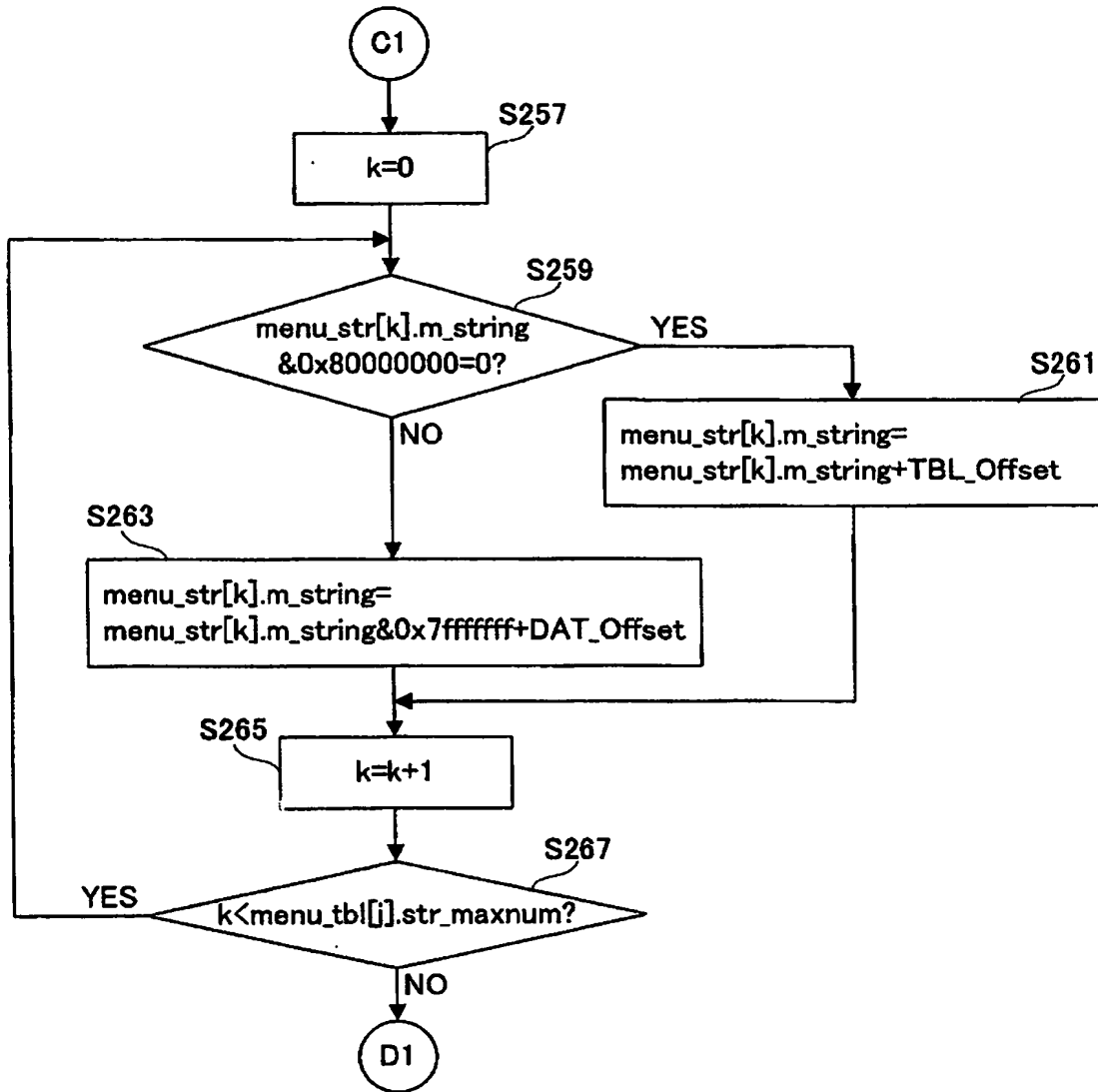
【図 26】



【図 27】

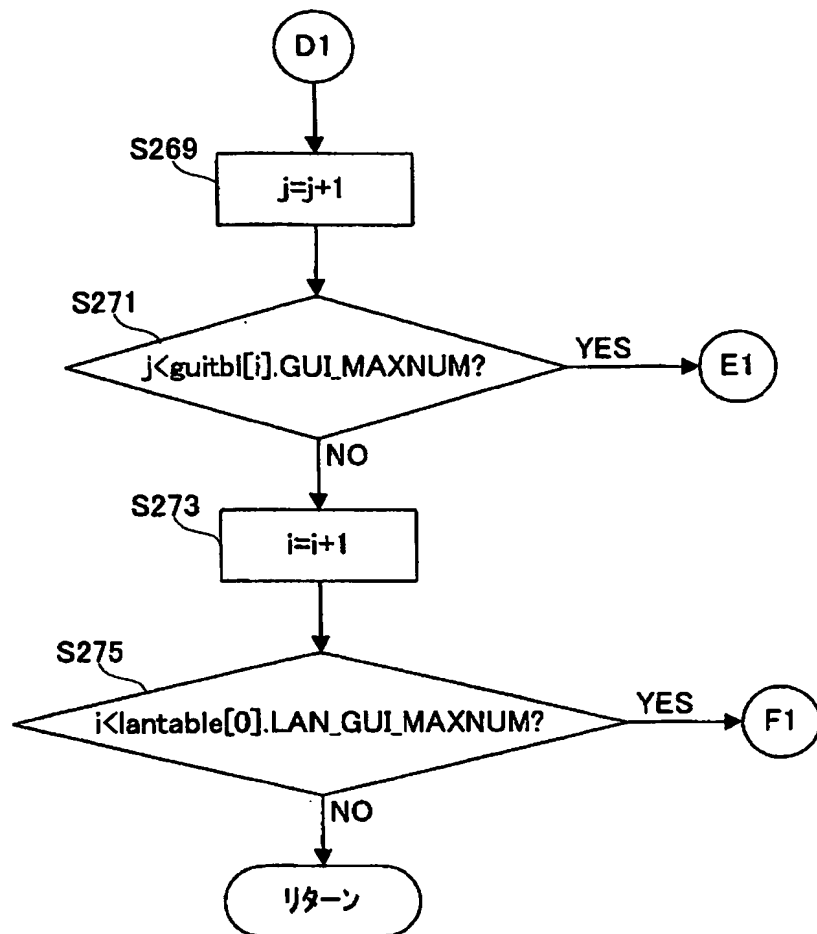


【図 28】

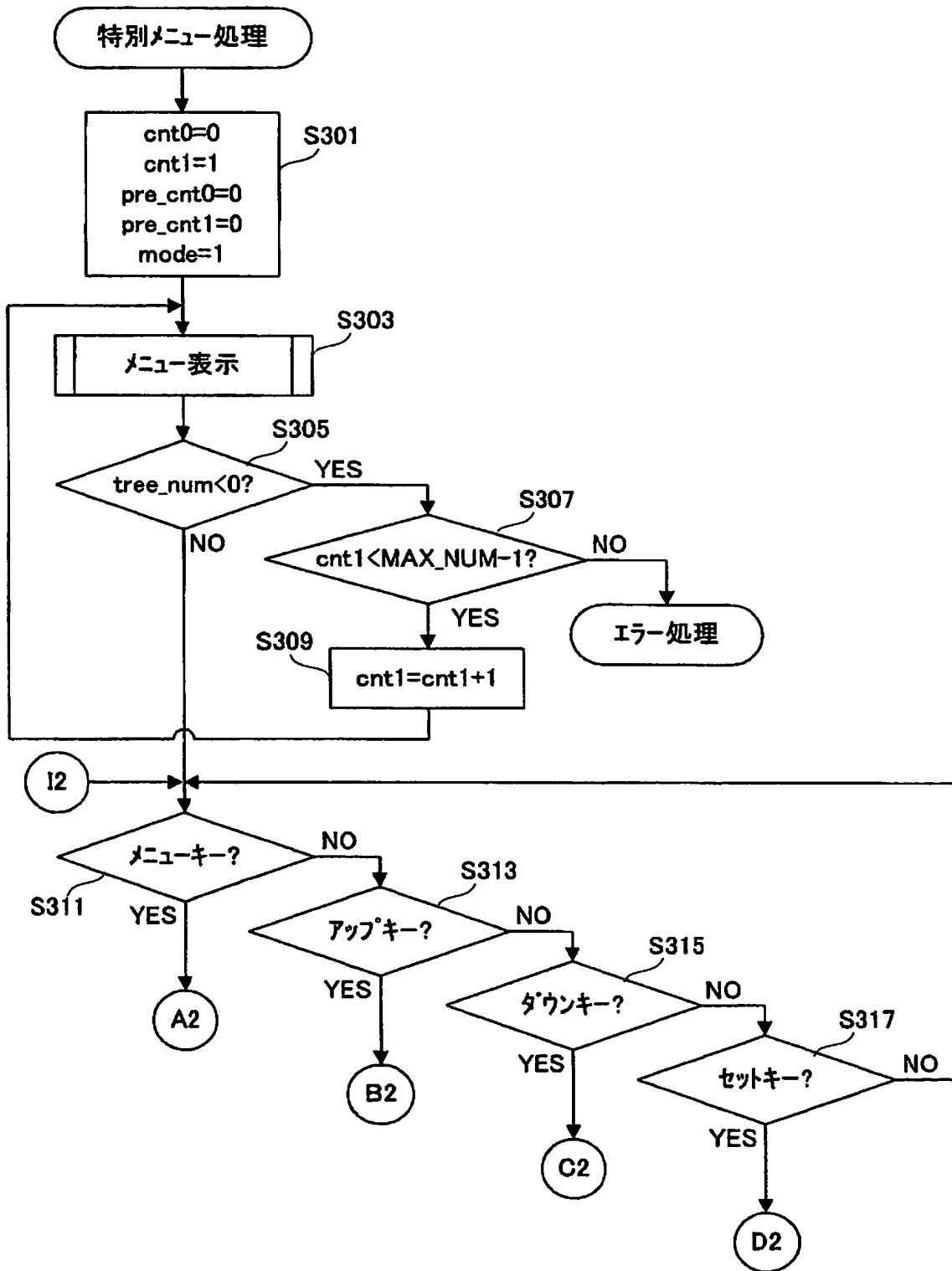




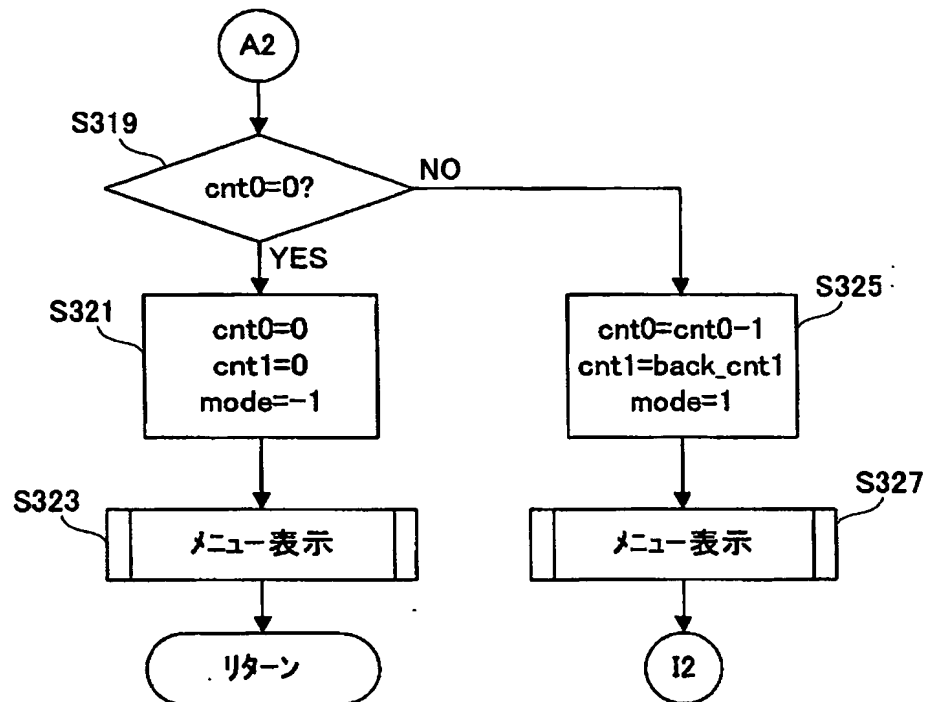
【図 29】



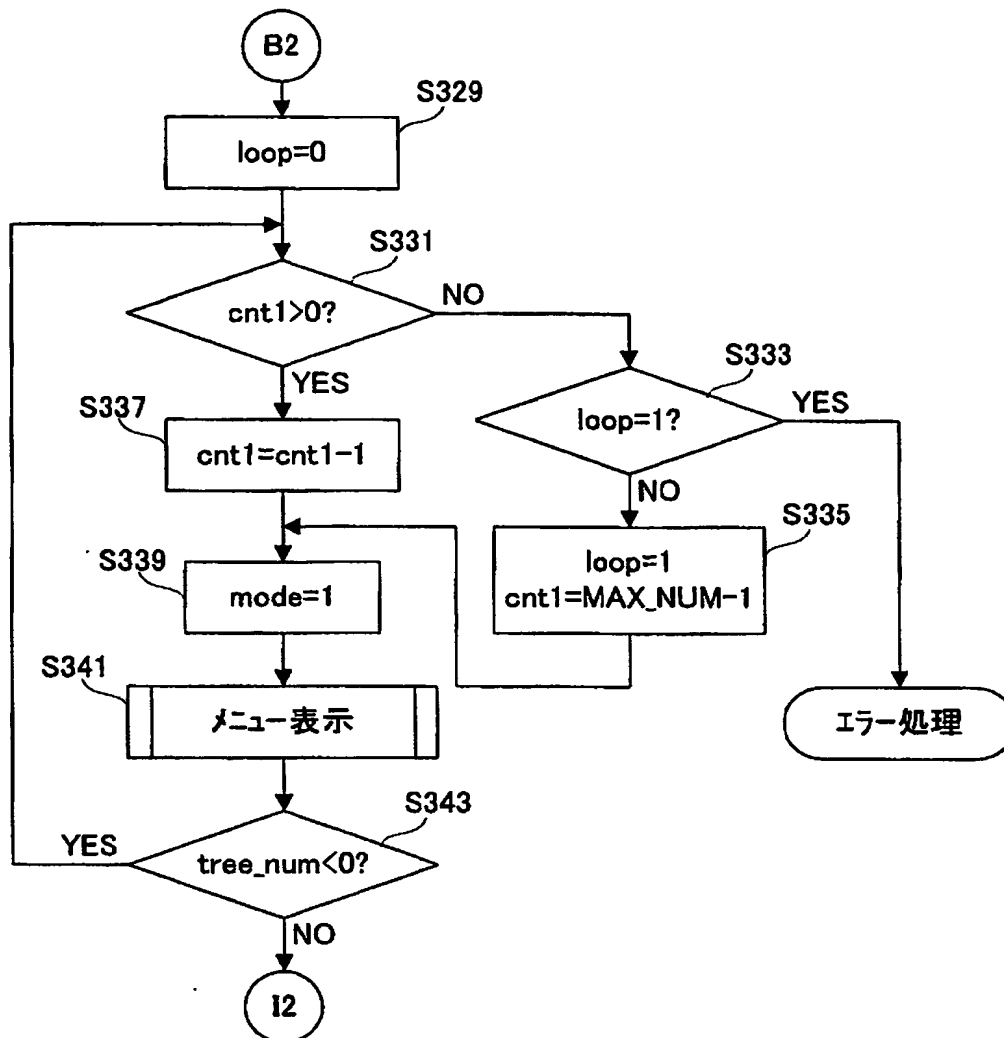
【図 30】



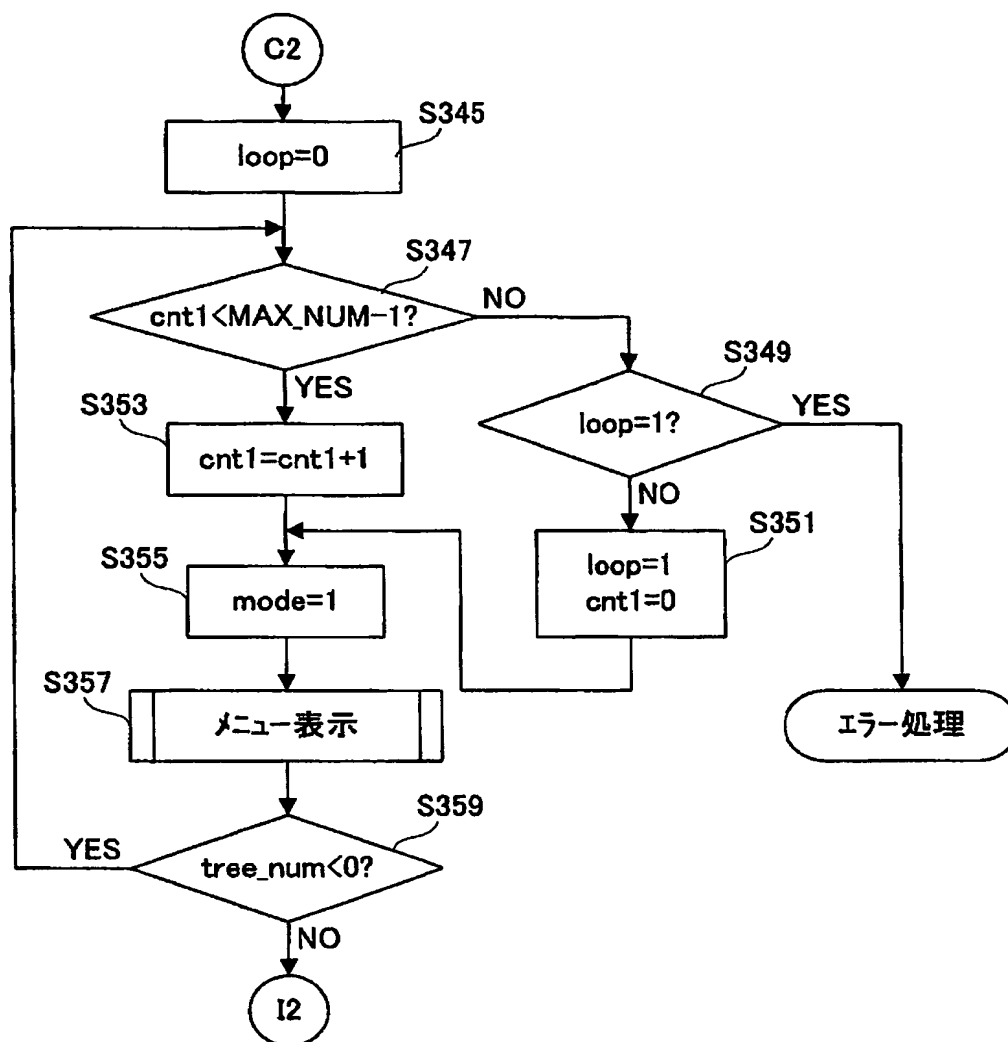
【図 31】



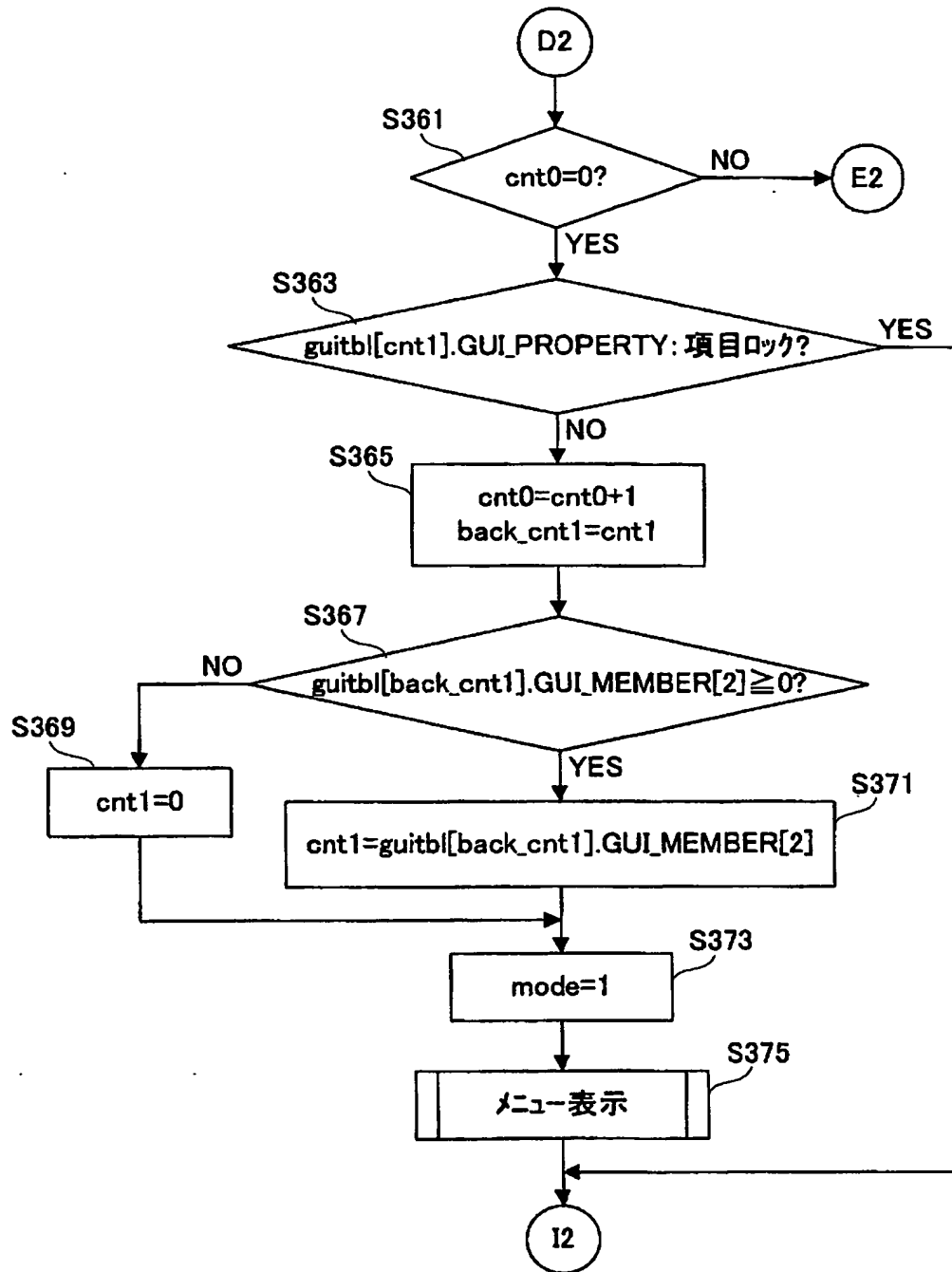
【図 32】



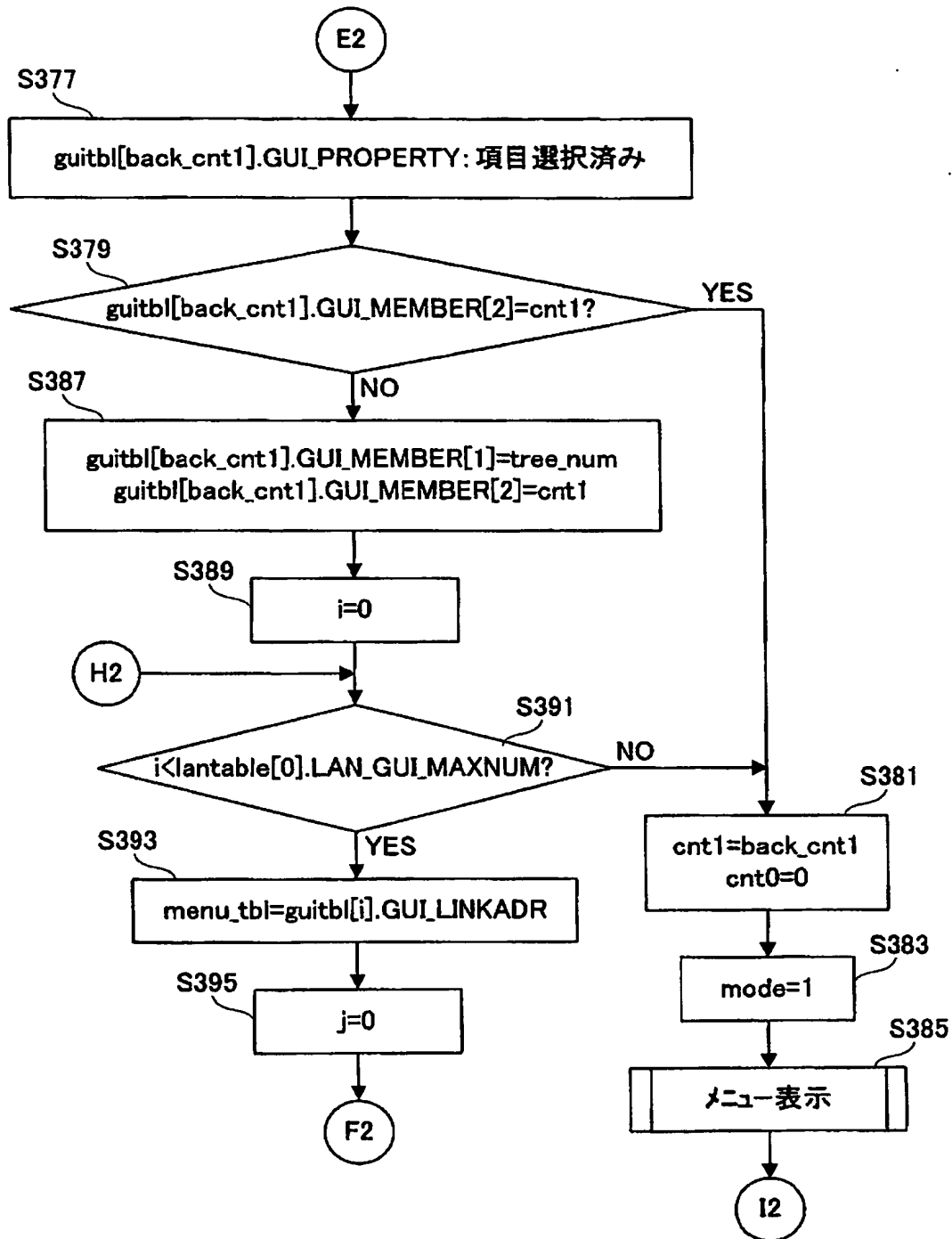
【図33】



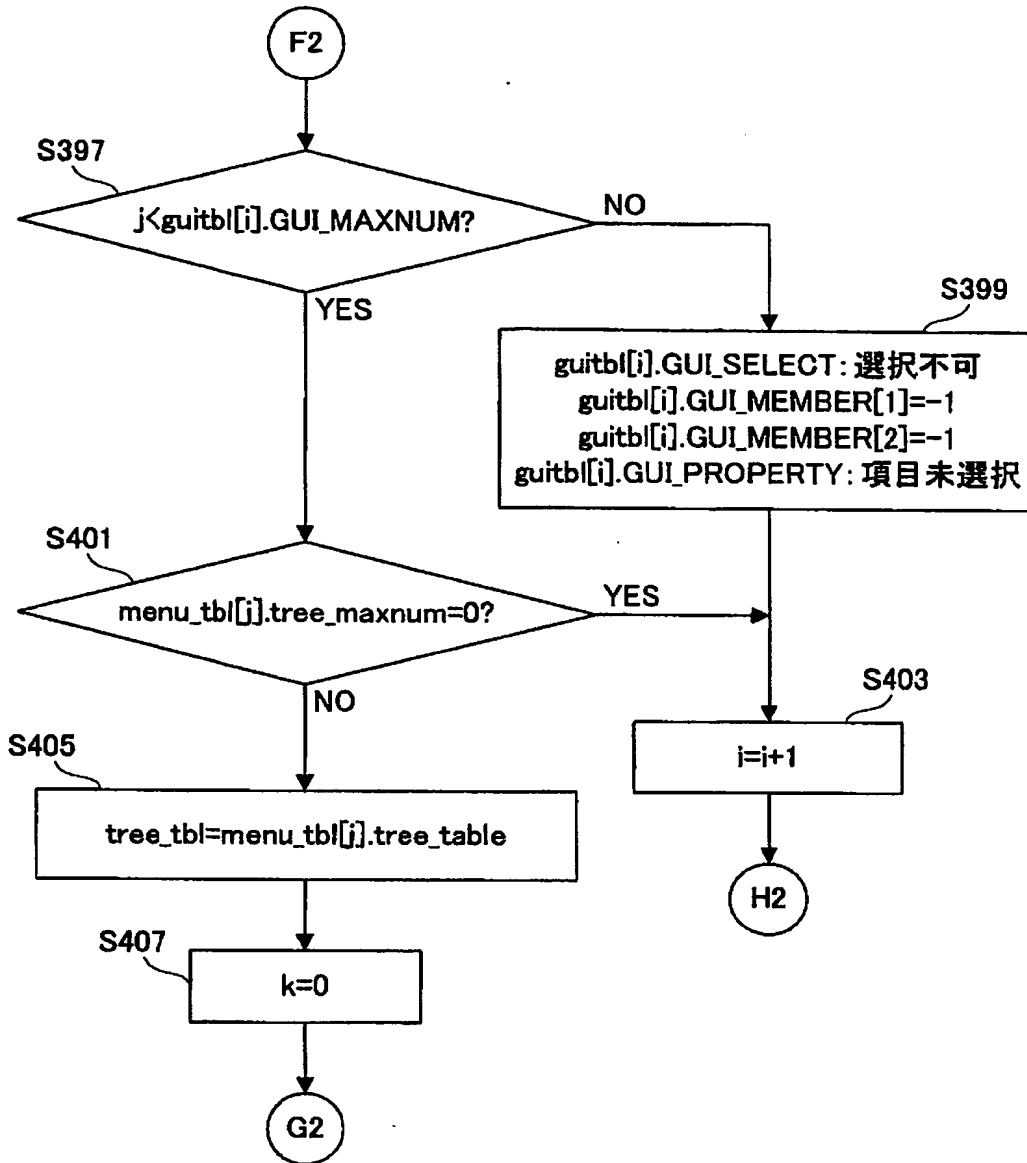
【図 34】



【図 35】

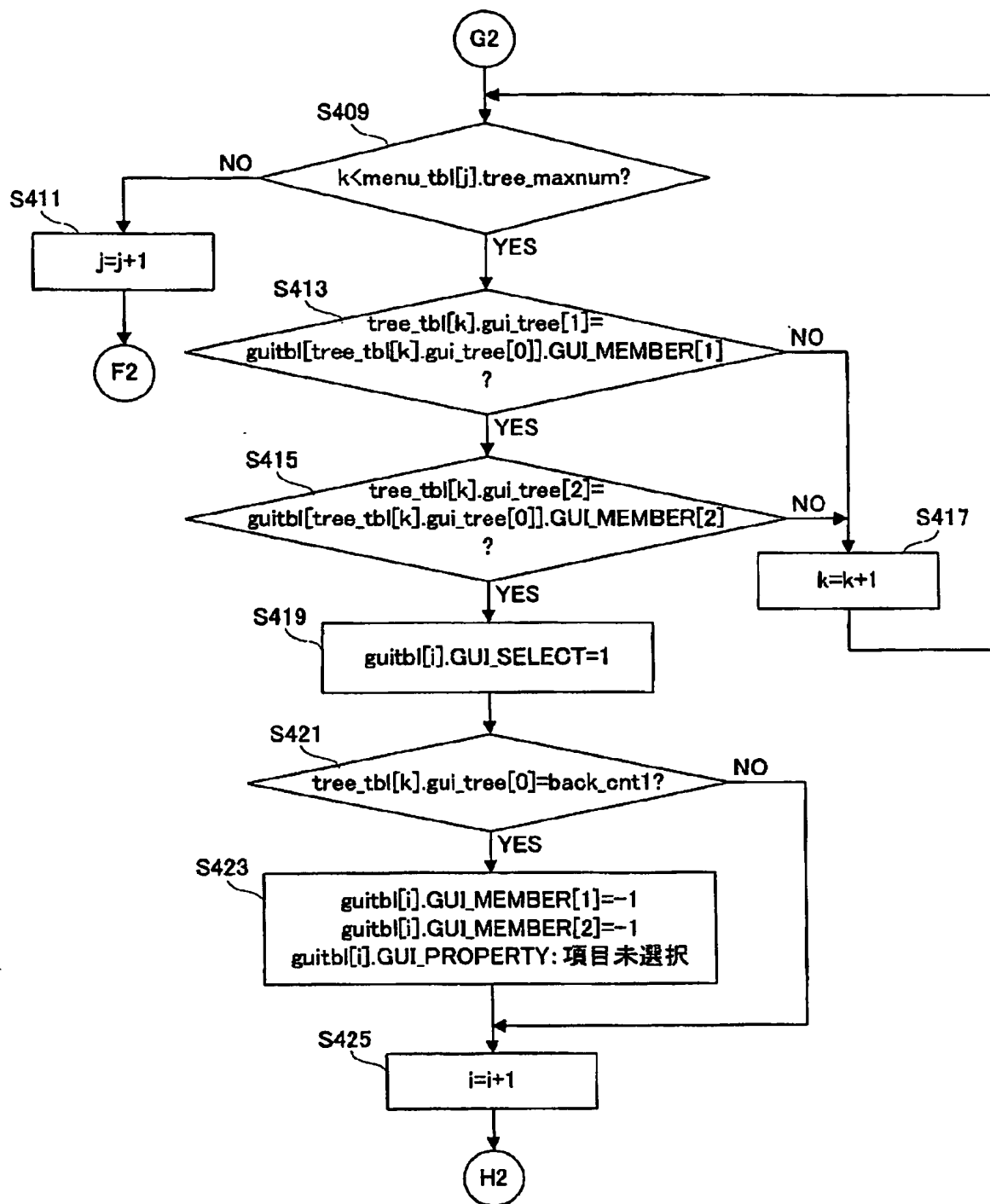


【図 36】

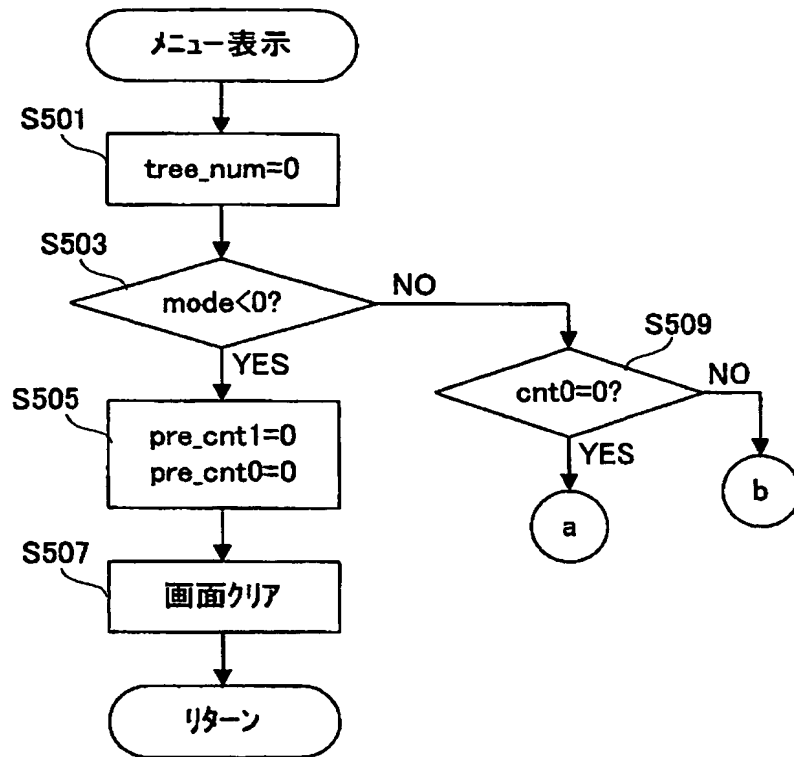




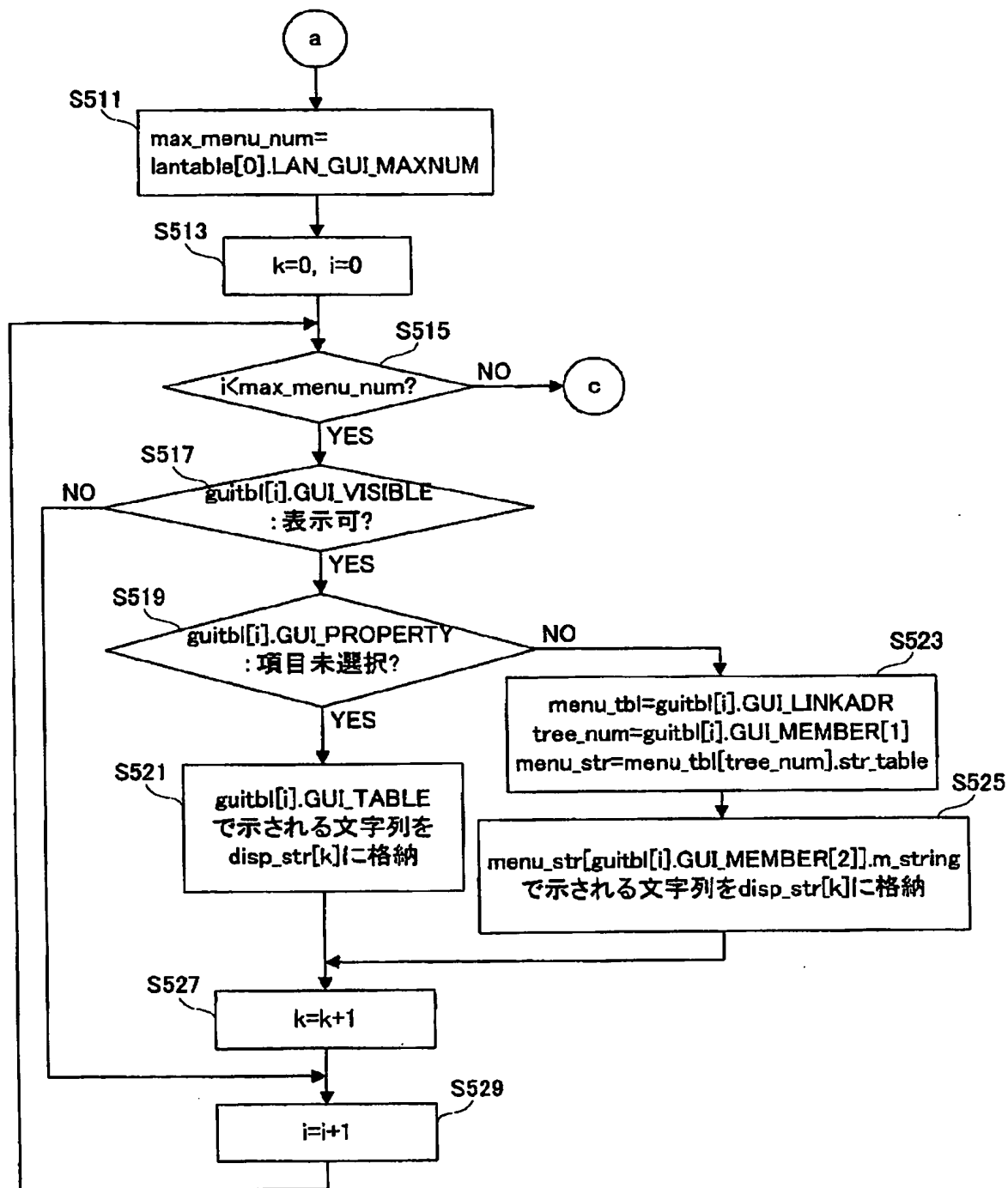
【図 37】



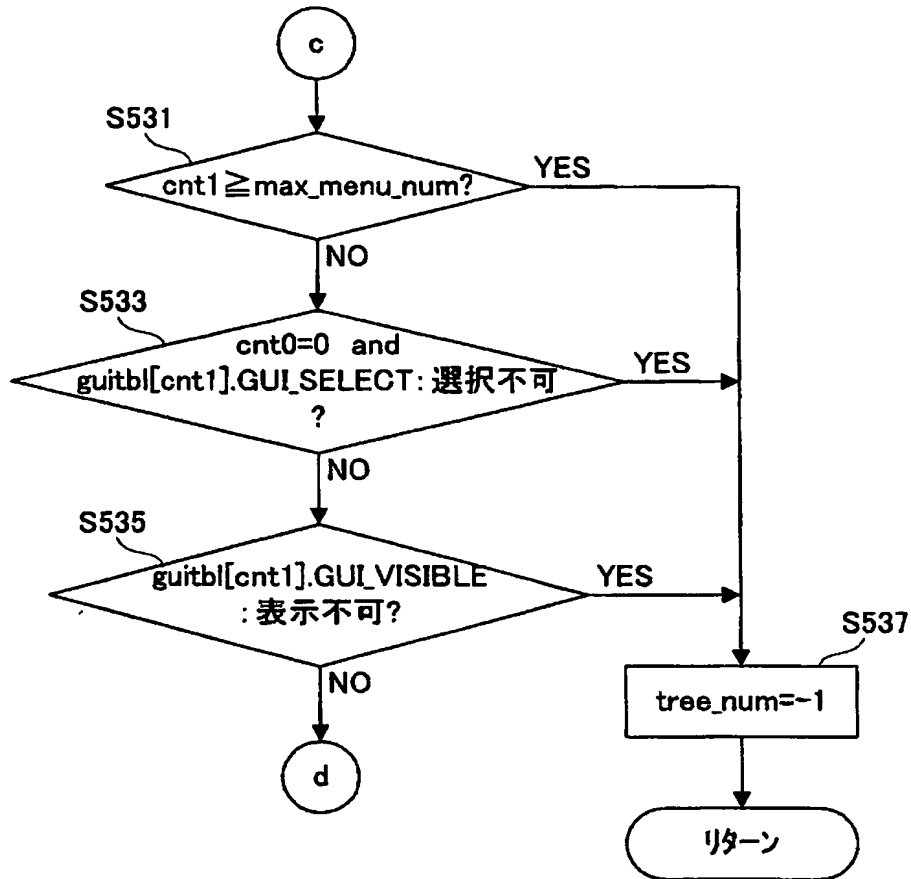
【図 38】



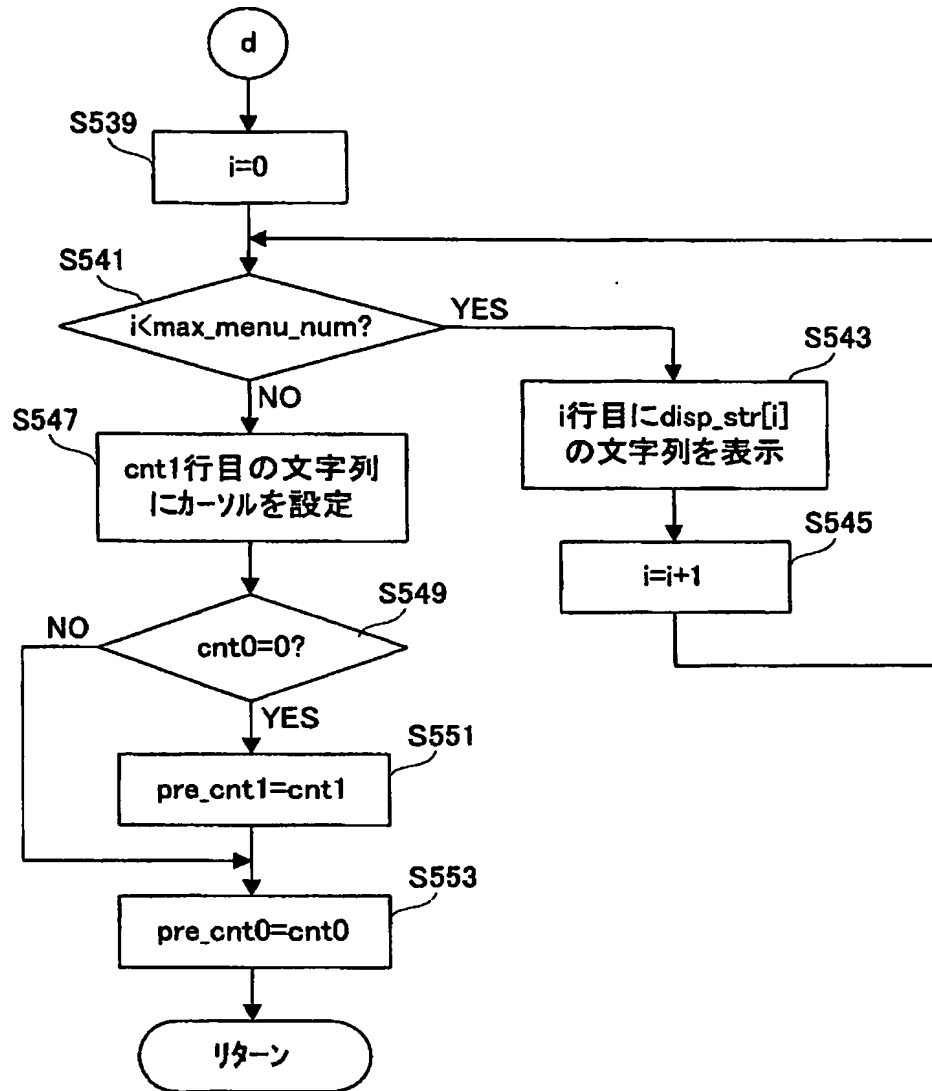
【図39】



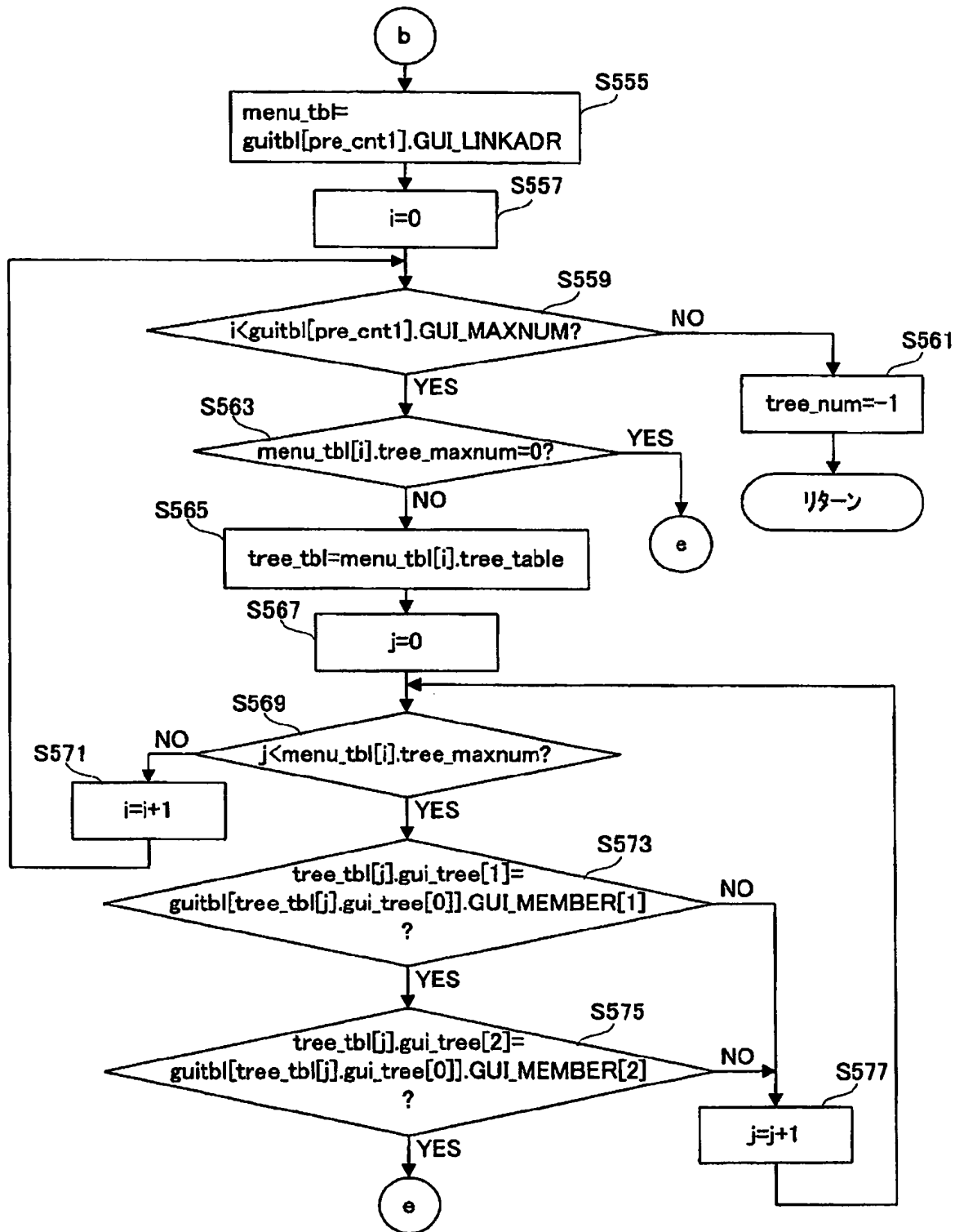
【図 40】



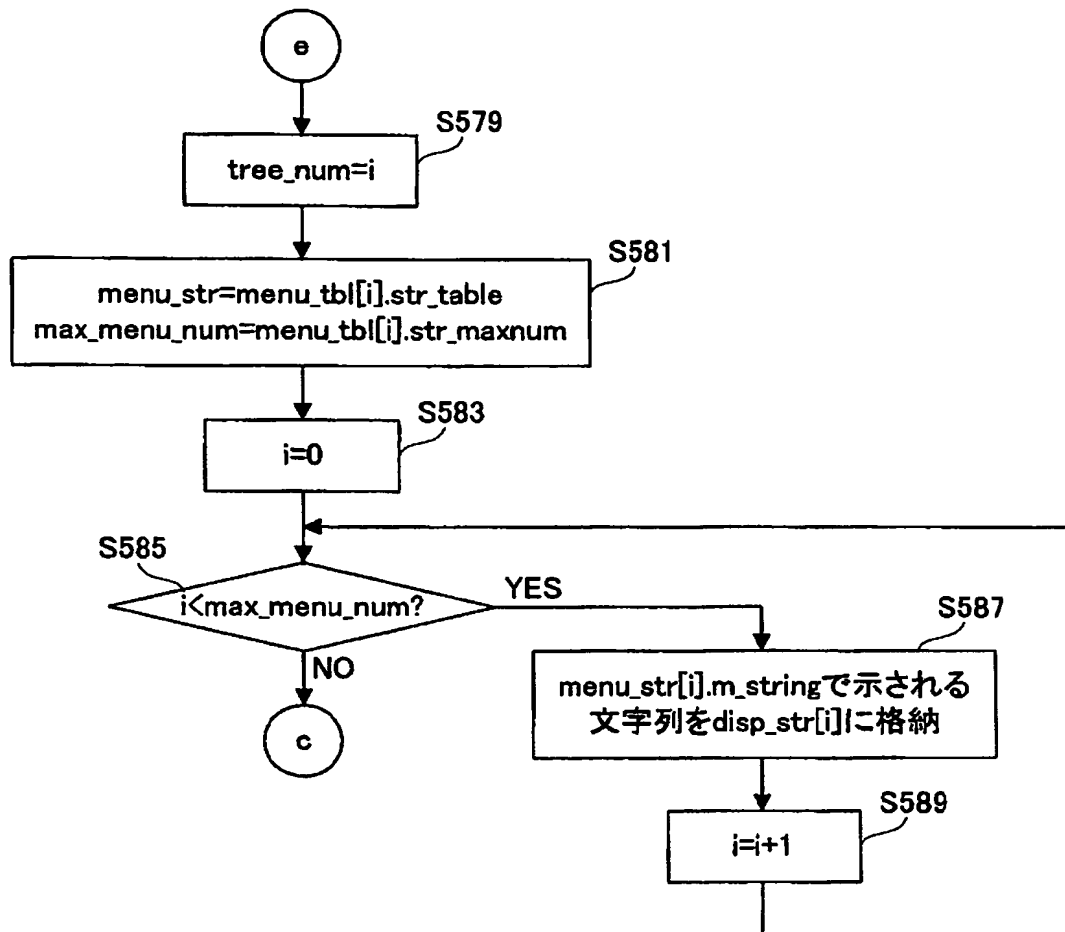
【図 4 1】



【図 4 2】



【図43】



## 【書類名】

要約書

## 【要約】

【構成】 表示制御テーブルは、G U I テーブル<guitbl>[0]～<guitbl>[N-1]とその各々に属するメニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[I-1]とを備える。G U I テーブル<guitbl>[0]～<guitbl>[N-1]は、表示処理を施される複数の大項目をそれぞれ管理する。また、メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[I-1]の各々は、表示処理を施される複数の小項目を管理する。メニューテーブル<menu\_\_tbl>[0]～<menu\_\_tbl>[I-1]の各々には、他のG U I テーブルの下で管理される小項目との依存関係を示すツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[L-1]が割り当てられる。所望の小項目が選択されると、このツリーテーブル<tree\_\_tbl>[0]～<tree\_\_tbl>[L-1]に基づいて、所望の小項目に依存する複数の小項目に表示処理が施される。

【効果】 デジタルカメラは、異なるメニュー画像を共通の手順で画面に表示することができる。

【選択図】 図 1 6



特願 2003-011857

出願人履歴情報

識別番号

[000001889]

1. 変更年月日

1993年10月20日

[変更理由]

住所変更

住 所

大阪府守口市京阪本通2丁目5番5号

氏 名

三洋電機株式会社

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**